

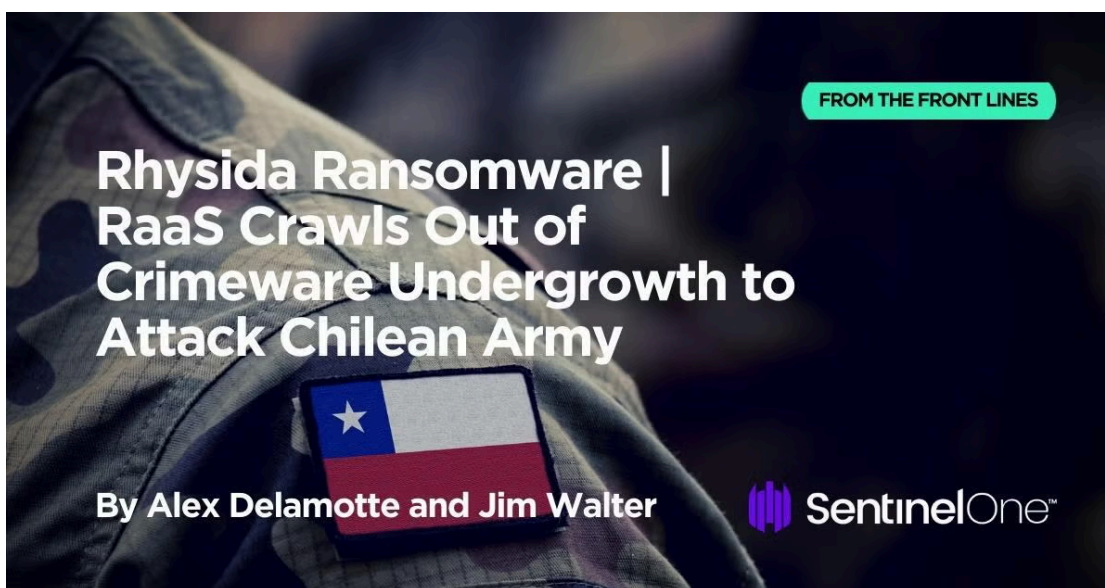
Rhysida Ransomware | RaaS Crawls Out of Crimeware Undergrowth to Attack Chilean Army

By Alex Delamotte & Jim Walter

Published: 2023-06-29 · Archived: 2026-04-05 19:33:28 UTC

The Rhysida ransomware-as-a-service ([RaaS](#)) group has gone from a dubious newcomer to a fully-fledged ransomware operation. Despite the developer's partial implementation of some features, the group emerged onto the scene at the end of May with a high-profile attack against the Chilean Army, continuing the ongoing [trend](#) of ransomware groups targeting Latin American government institutions. On June 15, the group [leaked](#) the files stolen from the Chilean Army.

In this post, we provide a high-level overview of Rhysida ransomware activity and present technical details of the malware payloads, along with hunting rules and IoCs to aid threat hunters and security teams.



Recent Attacks Attributed to Rhysida

On May 29 2023, the Chilean Army [reported](#) that it had been the target of a cyberattack affecting the organization's internal network on Saturday, May 27. The attack was later attributed to Rhysida.

Strategically, the Rhysida group's attack against the army of Chile distinguishes this newcomer from the sea of ransomware newcomers. It should be noted that Rhysida is an apparently independent ransomware group: SentinelOne has not observed any overt connections to existing ransomware operations. As such, any potential geopolitical ramifications from attacking Chile's government are as yet unclear. This is not the first time a Chilean governmental organization has been compromised by a new ransomware family, as demonstrated by the ARCrypter [attack](#) in November 2022.

The attack was followed by the leaking of data belonging to the army on June 15th. Through the week of June 19 2023, Rhysida's leaks page displayed an influx of further victims, including multiple organizations in each of the following sectors:

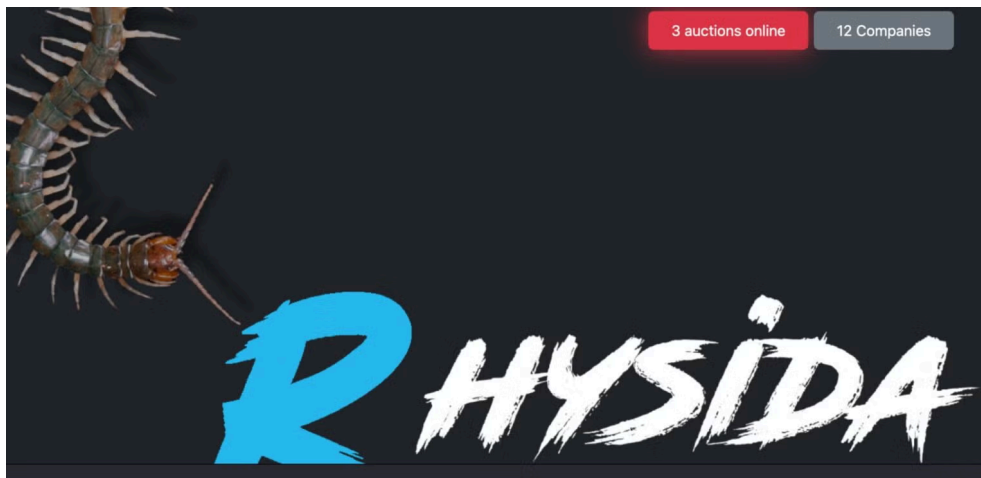
- Education
- Government
- Manufacturing
- Technology and Managed Service Providers (MSP)

Victims are distributed throughout Western Europe, North & South America, and Australia, loosely aligning the group's targeting with many ransomware operations that avoid targeting countries in Eastern Europe and Central Asia's

Commonwealth of Independent States. There are no Asian organizations posted at this time.

Operational Overview

The Rhysida ransomware group was first observed in May of 2023, following the emergence of their victim support chat portal, hosted via TOR (.onion). The name “Rhyshida” refers to a specific genus of centipede. This is also reflected in the ‘branding’ on their victim blog.



The genus Rhysida and the Rhysida ransomware logo

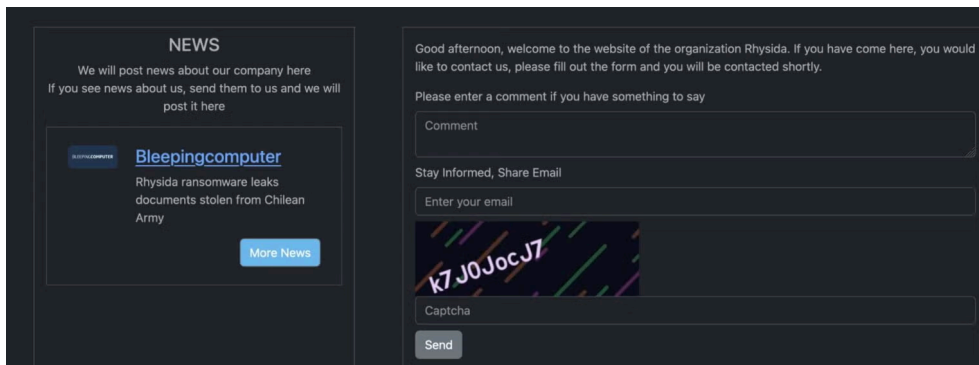
An Apache configuration status page reveals that the web server hosting the portal was first set up in March 2023. The group has since migrated their blog to a more ‘hardened’ instance of nginx, and these server configuration details and status are no longer visible. This move may have been prompted by the original IP address being exposed across various underground forums and markets.



Rhysida RaaS: Leakage of original blog IP address

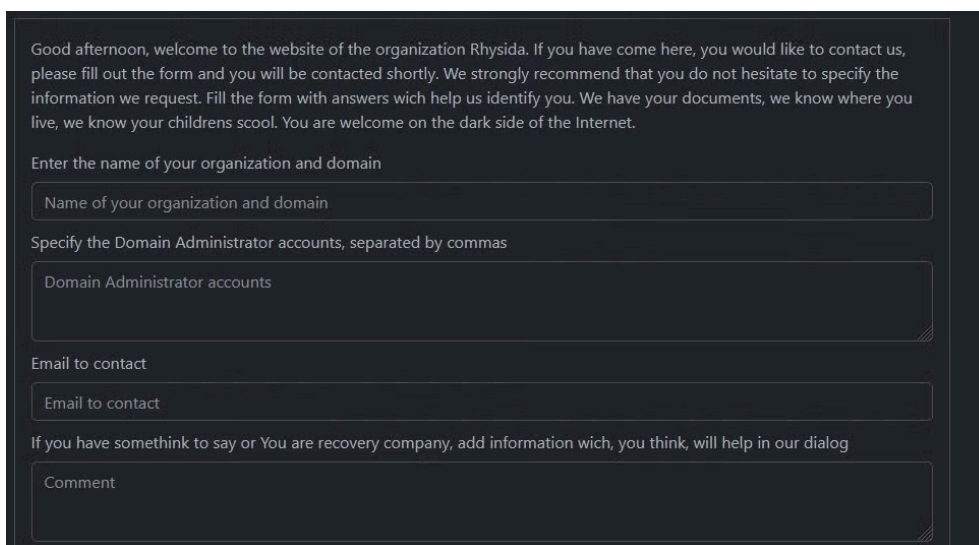
Rhysida is a privately marketed RaaS without known forum presence. The group positions themselves as a “cybersecurity team” who are doing their victims a favor by targeting their systems and highlighting the potential ramifications of the involved security issues. The group threatens victims with public distribution of the exfiltrated data, bringing them in line with modern-day multi-extortion groups.

The groups website also serves as a portal for Rhysida-centric news and media coverage, as well as details on how to contact the group should journalists, recovery firms or “fans” be inclined to do so.



Rhysida's 'communication portal'

Victims are instructed to contact the attackers via their TOR-based portal, utilizing their unique identifier provided in the ransom notes. Rhysida accepts payment in Bitcoin only, providing information on the purchase and use of BTC on the victim portal as well. Upon providing their unique ID to the payment portal, an additional form is presented that allows victims to provide additional information to the attackers, such as authentication and contact details.



Rhysida portal's additional details form

Technical Details

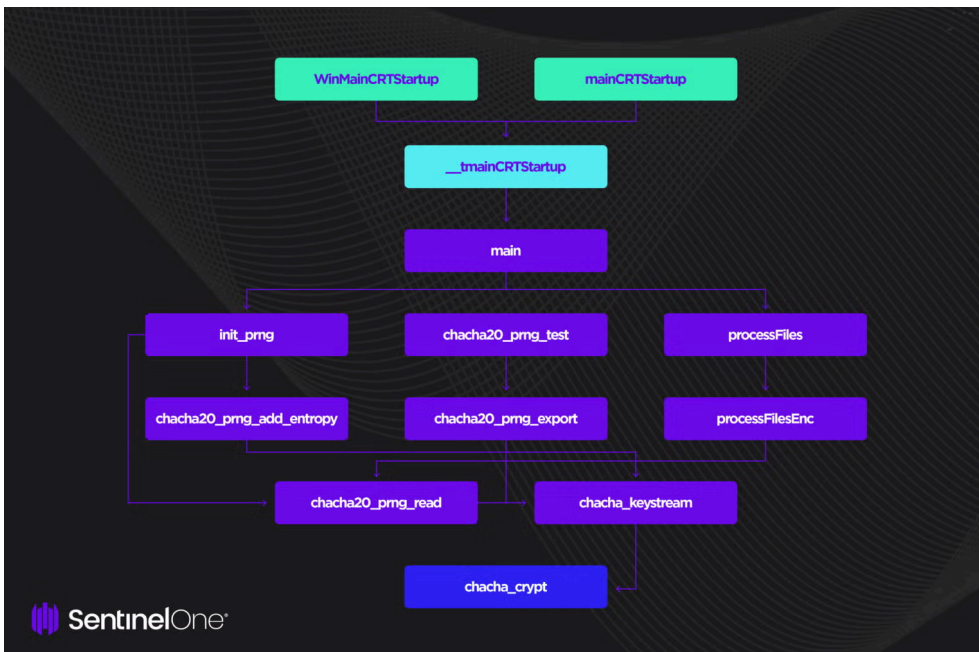
Rhysida is a 64-bit Portable Executable (PE) Windows cryptographic ransomware application compiled using MINGW/GCC. In each sample analyzed, the application's program name is set to `Rhysida-0.1`, suggesting the tool is in early stages of development.

A notable characteristic of the tool is its plain-text strings revealing registry modification commands.

Rhysida Encryption & File Processing

For encryption, Rhysida uses a 4096-bit RSA key with the ChaCha20 algorithm. Its `main` function initializes the ransomware's overall runtime, including encryption specifics. The `main` function contains several nested if-else conditions that handle arguments that specify different encryption implementations. The `processFileEnc` function contains code blocks for other encryption methods, including Rijndael, though the preceding functions are prefixed "test".

`processFileEnc` calls `init_prng`, which initializes the encryption routine's pseudo-random number generator that is passed to the `chacha_crypt` function.



Encryption function call graph from *main* to *chacha_crypt*

The `processFileEnc` function contains code that lists files and parses the current file name. Following encryption, Rhysida appends the `.rhysida` extension to the name of encrypted files.

After the encryption details are established, Rhysida enumerates files and folders connected to the system. The `main` function ends by calling PowerShell to delete the binary after encryption has completed.

```

79 }
80 else
81 {
82     for ( thread_i = 0; thread_i < PROCS; ++thread_i )
83     {
84         if ( init_prng(&prngs[thread_i], &PRNG_IDXS[thread_i]) )
85             goto LABEL_8;
86     }
87     if ( (unsigned int)rsa_import(_PUB_DER, _PUB_DER_LEN, &key) )
88     {
89         puts("ERROR rsa_import_key public");
90     }
91     else
92     {
93         err = register_cipher(refptr_aes_enc_desc);
94         if ( err )
95         {
96             v6 = (const char *)error_to_string((unsigned int)err);
97             printf("ERROR Unable to register aes_enc_desc cipher %s\n", v6);
98         }
99     }
100     else
101     {
102         CIPHER = find_cipher("aes");
103         if ( CIPHER == -1 )
104         {
105             puts("ERROR Cipher AES not found");
106         }
107     }
108     else
109     {
110         err = register_hash(refptr_chc_desc);
111         if ( err )
112         {
113             v7 = (const char *)error_to_string((unsigned int)err);
114             printf("ERROR register CHC hash %s\n", v7);
115         }
116     }
117     else
118     {
119         err = chc_register((unsigned int)CIPHER);
120         if ( err )
121         {
122             v8 = (const char *)error_to_string((unsigned int)err);
123             printf("ERROR binding AES to CHC %s\n", v8);
124         }
125     }

```

Rhysida *main* function encryption checks

Rhysida uses a file exclusion list to avoid encrypting certain files. This check occurs in the `isFileExcluded` function, which compares the current file extension against `exclude_extensions`, an array that contains the following excluded file extensions:

```
[ bat, bin,  
  cab, cmd, com, cur,  
  diagcab, diagcfg, diagpkg, drv, dll,  
  exe,  
  hlp, hta,  
  ico, ini, iso,  
  lnk,  
  msi,  
  ocx,  
  ps1, psm1,  
  scr, sys,  
  Thumbs-db,  
  url  
]
```

This function initializes two variables, `exclude_i` as 0 and `exclude_c` as 11, which iterate through the array of 27 excluded file extensions and the length of the current file name.

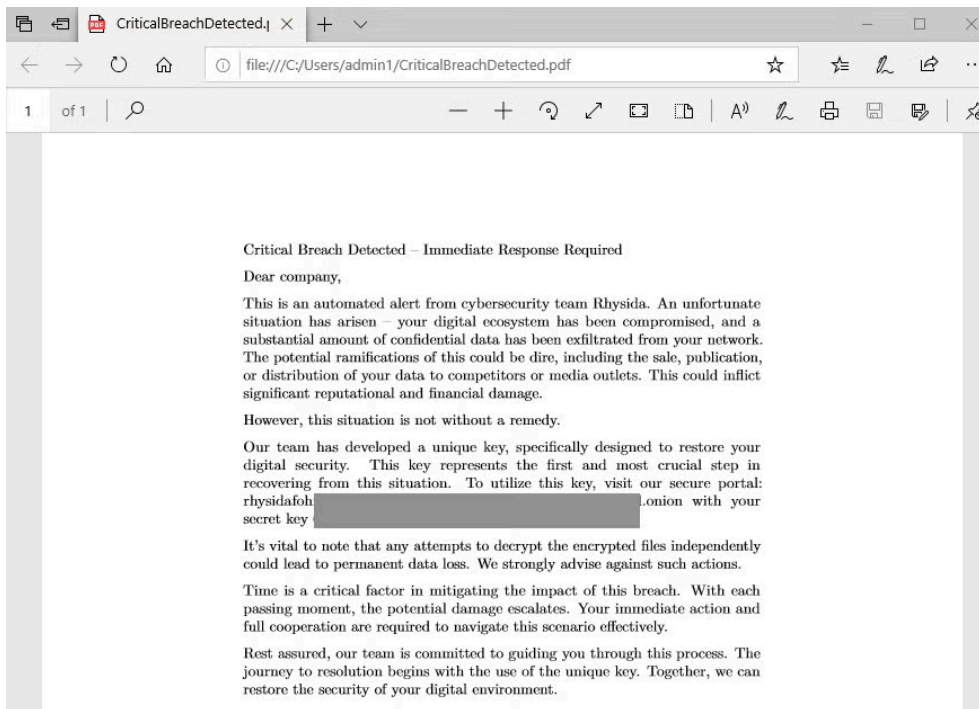
```
file_len = strlen(file_name);  
if ( CURRENT_TYPE_N == 1 )  
{  
  exclude_size = 0;  
  for ( exclude_i = 0; exclude_i <= 26; ++exclude_i )  
  {  
    for ( exclude_c = 11; exclude_c >= 0; --exclude_c )  
    {  
      if ( exclude_extensions[exclude_i][exclude_c] )  
      {  
        exclude_size = exclude_c;  
        break;  
      }  
    }  
    for ( exclude_ca = exclude_size;  
          exclude_ca >= 0  
          && (exclude_extensions[exclude_i][exclude_ca] == file_name[file_len - 1 - (exclude_size - exclude_ca)]  
            || exclude_extensions[exclude_i][exclude_ca] > 64  
            && exclude_extensions[exclude_i][exclude_ca] <= 90  
            && exclude_extensions[exclude_i][exclude_ca] + 32 == file_name[file_len - 1 - (exclude_size - exclude_ca)]  
            || exclude_extensions[exclude_i][exclude_ca] > 96  
            && exclude_extensions[exclude_i][exclude_ca] <= 122  
            && exclude_extensions[exclude_i][exclude_ca] - 32 == file_name[file_len - 1 - (exclude_size - exclude_ca)]);  
          --exclude_ca )  
    {  
      ;  
    }  
    if ( exclude_ca == -1 )  
      return 1;  
  }  
  for ( exclude_cb = _EXT_EXT_LEN - 1;  
        exclude_cb >= 0 && _EXT_EXT[exclude_cb] == file_name[exclude_cb - _EXT_EXT_LEN + file_len];  
        --exclude_cb )  
  {  
    ;  
  }  
  if ( exclude_cb == -1 )  
  {  
    return 1;  
  }  
}
```

Rhysida's `isFileExcluded` function

Extended features, beyond encrypting files, are still not present in current variations of Rhysida. The most recent of analyzed samples continue to lack commodity features like VSS Removal, multiple persistence mechanisms, process termination or unhooking.

Ransom Note & Victim Notification

Rhysida generates the ransom note as a PDF document. The content of the doc is embedded in the binary in clear text. This is a missed opportunity for the actors: PDF is a powerful document format that enables data to be encoded in many ways, often not in clear text. If the developer embeds the PDF object within the binary instead of constructing the PDF at runtime from unencrypted strings, Rhysida would evade string-based detection based on ransom note language.



Rhysida ransom note, *CriticalBreachDetected.pdf*

Rhysida's `setBG` function is designed to create a new image, write it to `C:\Users\Public\bg.jpg`, and run registry modifications via `cmd.exe` to change the wallpaper and prevent the victim's ability to change it. During SentinelOne' analysis, this process did not execute successfully and the JPG is not written to disk.

```

126         stbtt_GetCodepointBitmapBox(&font_info, _NOTE_TXT[letter_i], scale, scale, &c_x1, &c_y1, &c_x2, &c_y2);
127         y = ascent + c_y1;
128         v4 = (float);
129         v5 = (float)((float)(v4 + roundf(v3)) + (float)(img_width * y)
130             + (float)(img_width * cur_line_n * (line_gap + line_height)));
131         byte_offset = (int)(float)((float)(img_width * padding_y) + v5);
132         stbtt_MakeCodepointBitmap(
133             &font_info,
134             &image_data[byte_offset],
135             c_x2 - c_x1,
136             c_y2 - c_y1,
137             img_width,
138             scale,
139             scale,
140             _NOTE_TXT[letter_i]);
141         v6 = (float);
142         x = (int)(float)(roundf(v5) + v6);
143         kern = stbtt_GetCodepointKernAdvance(&font_info, _NOTE_TXT[letter_i], _NOTE_TXT[letter_i + 1]);
144         v7 = (float);
145         x = (int)(float)(roundf(v5) + v7);
146     }
147 }
148 }
149 }
150 }
151 sprintf(image_file, "C:/Users/Public/bg.jpg");
152 stbl_write_jpg(image_file, img_width, img_height, 1, image_data, img_quality);
153 free(image_data);
154 system("cmd.exe /c reg delete \"HKCU\\Conttol Panel\\Desktop\" /v Wallpaper /f");
155 system("cmd.exe /c reg delete \"HKCU\\Conttol Panel\\Desktop\" /v WallpaperStyle /f");
156 system(
157     "cmd.exe /c reg add \"HKLM\\Software\\Microsoft\\Windows\\CurrentVersion\\Policies\\ActiveDesktop\" /v NoChangingWall
158     \"Paper /t REG_SZ /d 1 /f");
159 system(
160     "cmd.exe /c reg add \"HKLM\\Software\\Microsoft\\Windows\\CurrentVersion\\Policies\\ActiveDesktop\" /v NoChangingWall
161     \"Paper /t REG_SZ /d 1 /f");
162 system("cmd.exe /c reg add \"HKCU\\Control Panel\\Desktop\" /v Wallpaper /t REG_SZ /d \"C:\\Users\\Public\\bg.jpg\" /f");
163 system(
164     "cmd.exe /c reg add \"HKLM\\Software\\Microsoft\\Windows\\CurrentVersion\\Policies\\System\" /v Wallpaper /t REG_SZ /"
165     "d \"C:\\Users\\Public\\bg.jpg\" /f");
166 system(
167     "cmd.exe /c reg add \"HKLM\\Software\\Microsoft\\Windows\\CurrentVersion\\Policies\\System\" /v WallpaperStyle /t REG_SZ /d 2 /f");
168 system("cmd.exe /c reg add \"HKCU\\Control Panel\\Desktop\" /v WallpaperStyle /t REG_SZ /d 2 /f");
169 system("rundll32.exe user32.dll,UpdatePerUserSystemParameters");
170 }
    
```

Rhysida's `setBG` function

The `setBG` function pulls elements from the PDF ransom note and allocates them to a buffer, which then is inserted into a new JPG image. The developer misspelled Control Panel as `Conttol Panel` in two of the registry modification commands. We patched the binary to correct the spelling, but the wallpaper still did not change.

It is of note that this misspelling flaw persists across versions of Rhysida. Original versions (example: `69b3d913a3967153d1e91ba1a31ebed839b297ed`) compiled on May 15, 2023 as well as the sample associated with the Chilean Army attack (`358d4f4ec714359d589918cee1adad12ef231907`, compiled on May, 27, 2023) each contain this issue.

SentinelOne Protects Against Rhysida Ransomware

The SentinelOne Agent [detects](#) Rhysida ransomware and prevents execution and file encryption.

Ett fel inträffade.

Det går inte att köra JavaScript.

For details about Rhysida and other ransomware families, visit SentinelOne's Ransomware Anthology [page](#).

Conclusion

Rhysida represents an unusual combination of techniques that suggest the developer is thinking outside the confines of contemporary ransomware. Features like the PDF ransom note could be leveraged for enhanced stealth, while the wallpaper changing feature is quite obtrusive, though not yet functional.

There are hallmarks of a less seasoned actor, such as the unobfuscated registry modification and PowerShell commands seen throughout the program. However, these are cosmetic fixes. Time will tell whether the developer's choice to omit ubiquitous features, such as VSS copy deletion, will pay off or be supplemented through tools outside of the Rhysida application.

Indicators of Compromise (IOC)

SHA1	Description
69b3d913a3967153d1e91ba1a31ebed839b297ed	Rhysida PE first reported by MalwareHunterTeam
338d4f4ec714359d589918cee1adad12ef231907	Rhysida PE used in attack against Chilean Army
b07f6a5f61834a57304ad4d885bd37d8e1badba8	Rhysida PE, crashes during analysis

YARA Hunting Rule

SentinelOne is providing the following YARA rule that defenders can use to identify Rhysida ransomware binaries.

```
rule rw_rhysida {  
  
  meta:  
    author = "Alex Delamotte"  
    description = "Rhysida ransomware detection."  
    sample = "69b3d913a3967153d1e91ba1a31ebed839b297ed"
```

```
reference = "https://s1.ai/rhys"
strings:
  $typo1 = { 63 6D 64 2E 65 78 65 20 2F 63 20 72 65 67 20 64 65 6C 65 74 65 20 22 48 4B 43 55 51
  $cmd1 = { 63 6D 64 2E 65 78 65 20 2F 63 20 72 65 67 20 61 64 64 20 22 48 4B 43 55 5C 53 6F 66
  $cmd2 = { 63 6D 64 2E 65 78 65 20 2F 63 20 72 65 67 20 61 64 64 20 22 48 4B 4C 4D 5C 53 6F 66
  $byte1 = { 48 8D 05 72 AA 05 00 48 8B 00 8B 95 }
  $byte2 = { 48 8D 15 89 CF 03 00 48 89 C1 E8 F9 1C 03 00 44 }
condition:
  2 of them
}
```

Source: <https://www.sentinelone.com/blog/rhysida-ransomware-raas-crawls-out-of-crimeware-undergrowth-to-attack-chilean-army/>