REPORT PREPARED AUGUST 2022

Ransomware Scenario Emulation

This engagement was performed in accordance with the Statement of Work, and the procedures were limited to those described in that agreement. The findings and recommendations resulting from the assessment are provided in the attached report. Given the time-boxed scope of this assessment and its reliance on client-provided information, the findings in this report should not be taken as a comprehensive listing of all security issues.

This report is intended solely for the information and use of Illumio, Inc.

Table of Contents

Executive Report

Project Overview

Illumio, Inc. engaged Bishop Fox to measure the effectiveness of Illumio Core for blue teams to detect and contain a ransomware attack. The following report details the findings identified during the course of the engagement, which started on March 10, 2022.

GOALS

- Determine realistic metrics to measure the effectiveness of the Illumio Core product against a ransomware attack
- Develop an attack methodology based on real threat actors' tactics, techniques, and procedures (TTPs) to attack the test environment
- Execute attack emulations on the test environment based on the developed methodology to gradually measure the effectiveness of Illumio Core in detecting and responding to a ransomware attack

SCOPE

Illumio Core

DATES

Kickoff
03/10/2022

Active Testing
03/10/2022 – 04/05/2022

Report Delivery
04/18/2022

Summary of Testing

The assessment team performed a succession of attack emulations on a test environment to measure the effectiveness of Illumio Core against an active ransomware threat. The team developed a test environment mimicking the components of a real network, as well as a methodology mapped to the MITRE ATT&CK framework and based on real threat actors' tactics, techniques, and procedures (TTPs).

Using this test environment, the team ran a series of five attack scenarios and measured the following data points:

- Number of successfully infected or compromised hosts
- Time taken by the attacker to complete the scenario (whether the attack succeeded or was blocked by the security team)
- Number of TTPs that were successfully executed

These scenarios were executed by two Bishop Fox consultants, with one acting as the attacker (red team) and the other acting as the security team defending the test environment (blue team).

The main goal of these tests was to assess whether the Illumio Core product can accelerate the detection and response phase of a security incident like a ransomware attack and how it can complement existing solutions like endpoint detection and response (EDR) products.

Overall, the assessment team observed that the stricter the Zero Trust Segmentation (also known as microsegmentation) policy and enforcement modes, the faster it was for the blue team to detect and stop the ongoing attack. With Illumio deployed in a full application ring-fencing configuration, the blue team was able to contain the red team within ten minutes from the initial host compromise, compared to almost forty minutes in a passive configuration.

In terms of data collection, the team found Illumio's telemetry to be especially useful to cover some EDR blind spots, where attacker activities were not properly detected by the preconfigured EDR alerts. In a particular scenario where the red team performed more evasive maneuvers, the team properly identified a suspicious traffic pattern using Illumio's telemetry combined with EDR alerts.

To conclude, the team found that Illumio offers a range of capabilities that significantly improve an organization's ability to detect, contain and proactively limit the available attack surface:

- Zero Trust Segmentation (ZTS) can be applied to effectively isolate compromised hosts during an active attack.
- ZTS can be used proactively to ring-fence entire environments and applications, drastically reducing the pathways available for exploit through lateral movement. This is exemplified by the attack being made ineffective within 10 minutes when full app ring-fencing policies were enforced, compared to the attack being active for almost 2.5 hours when no segmentation was in place.
- The context-rich traffic visibility provided by the Illumio ZTS platform can complement data available from existing EDR, endpoint protection platform (EPP), or extended detection and response (XDR) solutions to provide blue teams with more coverage and to further enhance detection and response.

Attack Scenario Results

The assessment team developed scenarios to measure the effectiveness of the Illumio Core product. Details about these scenarios and their results can be found in the Assessment Report section of the report.

SUMMARY OF ATTACK SCENARIOS

SCENARIO	COMPROMISED HOSTS	TIME TO COMPLETE	ATTACK STOPPED?	SUCCESSFUL TTPs
Scenario 1: Control test (Illumio not deployed)	16 of 16	2 hours, 28 minutes	No	26 of 26
Scenario 2: Detection and response	2 of 16	38 minutes	Yes	12 of 13
Scenario 3: Preconfigured static protection	2 of 16	24 minutes	Yes	7 of 9
Scenario 4: Full application ring-fencing	1 of 16	10 minutes	Yes	6 of 8

In the table above, the Time to Complete measurement represents either the time it took for the attacker to meet their goal of compromising all the test environment, or the time it took for the defender to completely stop the attack.

The Successful TTPs column represents the number of TTPs that were successfully executed (not blocked by security solutions) out of the total number of TTPs that the attacker tried to run before the attack was blocked or the attacker goals were achieved.

TIME TO COMPLETION (MINUTES)

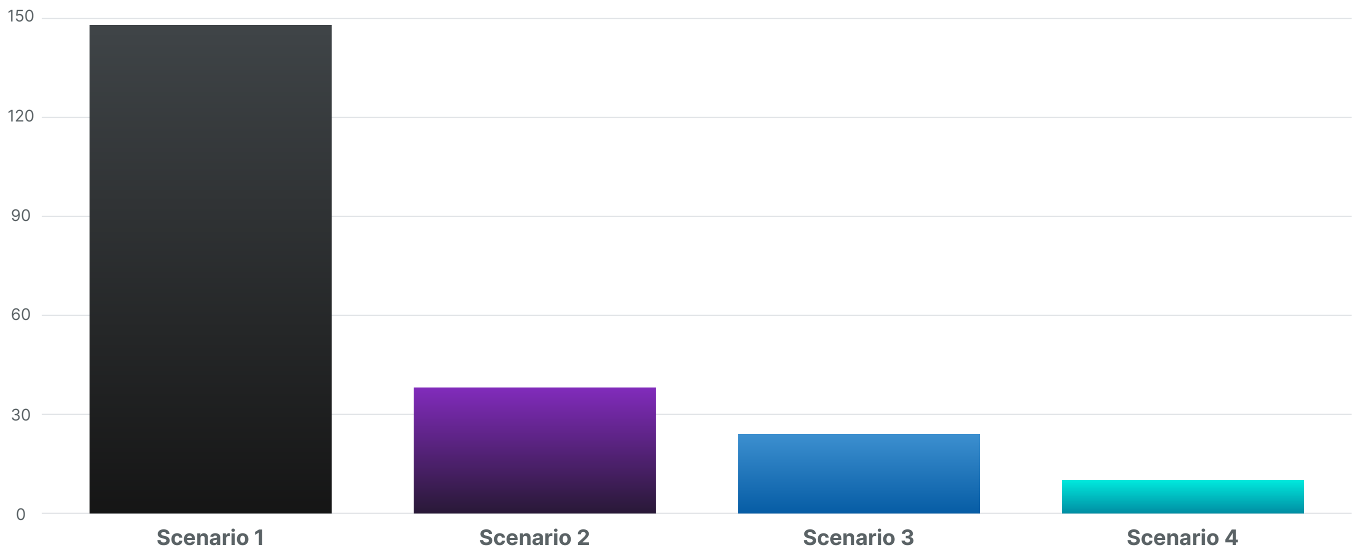


FIGURE 1 - Time to completion in minutes

The above figure summarizes the results from the table on the previous page, highlighting the completion time against the time it took for the blue team to detect the attacks. The next figure highlights the number of TTPs attempted for each scenario:

TTPs ATTEMPTED

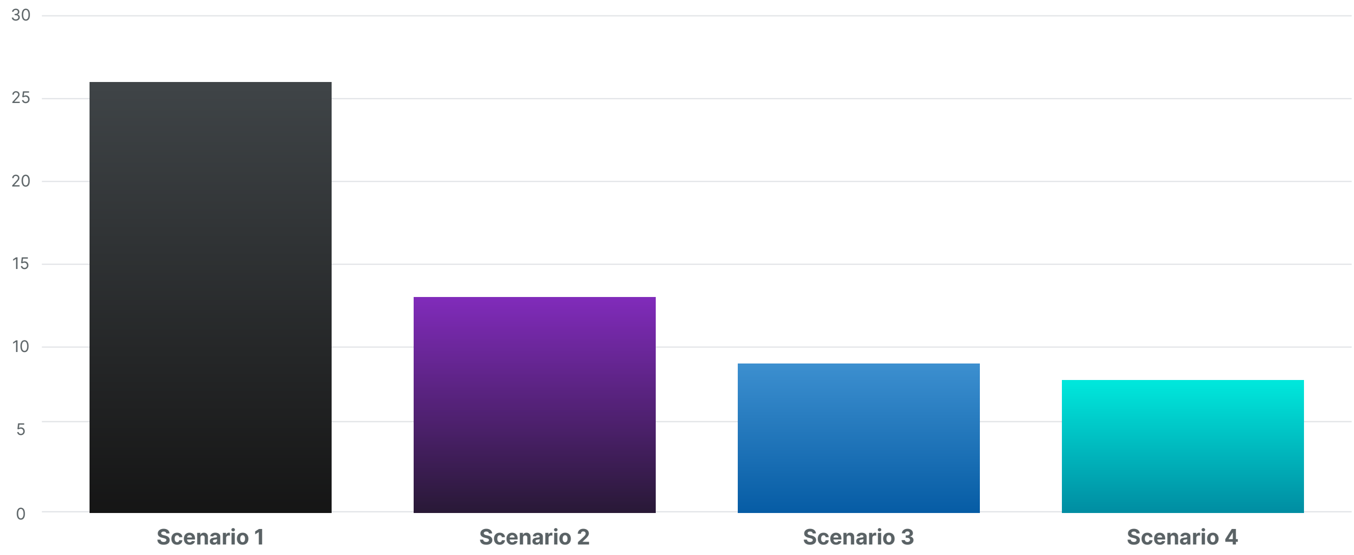


FIGURE 2 - Number of TTPs attempted for each scenario

The assessment team observed that the more microsegmentation was applied, the less time it took to block the attacks, resulting in a lesser number of TTPs executed by the attacker.

Test Environment Setup

To ensure the test environment could easily be set up between each scenario execution, the assessment team opted for an infrastructure-as-code solution. The team based this environment on the Splunk Attack Range open source project, which they modified to include more hosts and deploy a more complete Active Directory configuration.

The test environment was comprised of the following resources:

- Five Windows Server 2019 instances representing hosts in a corporate network
- Five Windows Server 2019 instances representing hosts in a staging network
- Five Windows Server 2019 instances representing hosts in a production network
- One Windows Server 2019 acting as a domain controller
- One Ubuntu 18.04 server running a Splunk server

All Windows instances were running a Splunk Universal Forwarder agent and a System Monitor (Sysmon) service configured with the default Splunk Attack range configuration. These instances were also deployed with the default configuration of Nextron Systems' Aurora EDR agent, including the default set of Sigma rules. All Windows hosts had the following remote administrative services enabled:

- Windows Remote Management (WinRM)
- Remote Desktop Protocol (RDP)

On top of that configuration, the Illumio VEN agent was installed during instance provisioning.

This configuration ensured that anyone could be able to reproduce the results the team observed with free and available software, without requiring specific licenses (apart from the Illumio license).

The complete environment was deployed using a combination of Terraform scripts and Ansible playbooks to ensure every run could be automatically deployed. Everything was deployed on AWS.

Each host belonging to a specific network segment (Corporate, Staging, Production) was put in a dedicated organizational unit (OU) in Active Directory. For each OU, a dedicated group was created and granted local administrative access to all hosts belonging to the same OU. This setup allowed the team to create administrative users for each segment, incrementally increasing the difficulty to compromise the complete environment.

For lateral movement, the team planted credentials in text files that were accessible in file shares exposed by the instances in each network segment. Specifically, credentials to access the Staging environment were stored on the corporate network, credentials to access the Production environment were stored on the Staging network, and a domain administrator's credentials were stored on the Production network. User access controls were enforced to restrict the shared file access to administrative users in each environment, forcing the attacker to pivot horizontally to progress further in their attack.

Attack Methodology

To conduct the attack emulations, the assessment team extracted relevant TTPs from the MITRE ATT&CK and PRE-ATT&CK frameworks, based on the test environment expectations. To accurately replicate real-world attacks, the assessment team created playbooks based on known techniques of active ransomware threat groups such as Conti.

The attacker's goal in the scenario was primarily identification of available assets, lateral movement, privilege escalation within the environment, and the deployment of ransomware across the domain-joined systems. For the complete list of TTPs, please refer to Appendix A.

Based on the selected TTPs, the team determined the following approach to the testing activities:

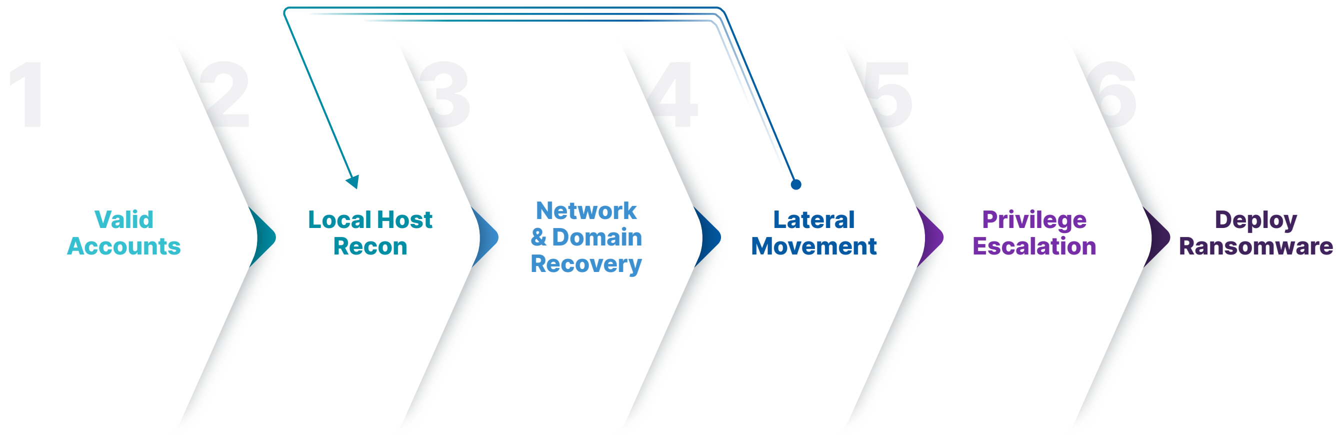


FIGURE 3 - Ransomware methodology

This approach was used for all of the test cases. For each host the red team successfully pivoted to, the team recorded the accessible machines and shares available with the gained access. Initial access in each scenario started with the attacker using RDP and valid credentials, emulating an exposed network service and credentials gained from previous attacks or breaches.

Attack Scenarios

To gradually measure the effectiveness of Illumio Core, the assessment team performed a total of five different attack emulations:

- Control test – Illumio not deployed
- Detection and response
- Preconfigured static protection
- Full application ring-fencing
- Preconfigured static protection with dynamic updates

The control test and subsequent scenarios were all conducted against the same network of 16 domain-joined machines, with increasingly complex controls and segmentation configurations. Between each execution, the test environment was destroyed and rebuilt to ensure no artifacts from previous runs were present.

The sections below describe the setup and observable results from each of these tests, including the metrics observed by Bishop Fox during the executed activities.

SCENARIO 1: CONTROL TEST

Setup

This scenario was a control test with no Illumio capabilities deployed, to get baseline measurements for the attack. In this scenario, the network was flat, without any network segmentation.

All system logs captured by Sysmon and the Aurora EDR agent were forwarded to the centralized Splunk instance, allowing the team to analyze the attack as it occurred.

Test Results

The red team followed the methodology described in the previous section to perform reconnaissance, credential gathering, lateral movement, data exfiltration, privilege escalation, and the ultimate deployment of ransomware targeting internal network shares.

In this round of testing, the red team attacked the target network without Illumio installed in order to establish a baseline of the environment and correct any issues with the emulated attacker playbook and tooling. To start the scenario, the red team connected to the RDP service on the machine `corp-win-serv-0` using the account `ATTACKRANGE\CORPADMIN`. Next, the team uploaded a Sliver post-exploitation agent in order to gain remote command and control over the compromised host without relying on RDP sessions.

The team continued by creating a staging directory for data exfiltration in the path C:\ProgramData\TempData. With the staging directory in place, the team continued with common local host and network discovery used by the Conti ransomware group. This provided the team with the following information:

- The domain controller
- Access permissions of user corpadmin
- A list of domain and local administrators
- Group Policy Objects

The following excerpt shows a portion of the results from the group policy discovery:

```
$ gpresult /R
OS Configuration:             Member Server
OS Version:                   10.0.14393
Site Name:                    N/A
Roaming Profile:              N/A
Local Profile:                C:\Users\corpadmin
Connected over a slow link?: No

USER SETTINGS
--------------
    CN=admin,OU=Users,OU=Corporate,DC=attackrange,DC=local
    Last time Group Policy was applied: 3/28/2022 at 6:44:43 PM
    Group Policy was applied from:      win-dc.attackrange.local
    Group Policy slow link threshold:   500 kbps
    Domain Name:                        ATTACKRANGE
    Domain Type:                        Windows 2008 or later
…omitted for brevity…
```

FIGURE 4 - Group policy discovery

The red team continued network discovery by searching the domain controller's SYSVOL share for Group Policy Preferences (GPP) that contained cached credentials by injecting a .NET assembly of Net-GPPPassword_dotNET_v4.exe:

```
assembly = Net-GPPPassword, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
@[]
Processing files in \\ATTACKRANGE.LOCAL\sysvol\ATTACKRANGE.LOCAL\policies\
Finished processing!
```

FIGURE 5 - GPP password search

These activities were detected by the blue team thanks to an EDR alert, triggering an investigation to identify the compromised hosts. However, no actions were taken to actively stop the attack.

As the GPP files did not reveal any credentials, the team continued by conducting Kerberoasting attacks against users in the domain. Kerberoasting allows for the extraction of account credential hashes from Active Directory that can be subsequently used in offline password cracking attacks. The team successfully pulled the hashes for three user accounts using an injected .NET Rubeus assembly, as shown below:

```
[*] Action: AS-REP roasting
[*] Target Domain          : attackrange.local
[*] SamAccountName         : PENNY_MAYO
[*] DistinguishedName      :
CN=PENNY_MAYO,OU=Users,OU=Corporate,DC=attackrange,DC=local
[*] Using domain controller: attackrange.local (10.0.1.14)
[*] Building AS-REQ (w/o preauth) for: 'attackrange.local\PENNY_MAYO'
[+] AS-REQ w/o preauth successful!
[*] AS-REP hash:

      $krb5asrep$PENNY_MAYO@attackrange.local:F3
      $krb5asrep$TRUMAN_BURNS@attackrange.local:B3
      $krb5asrep$CHANCE_KEITH@attackrange.local:9B
```

FIGURE 6 - Kerberoasting results

The team extracted the hashes and attempted to crack them offline using Hashcat and common password rule lists but was unsuccessful in recovering the cleartext passwords. The red team continued with network enumeration by injecting a .NET assembly of SharpView to gather a list of domain-joined machines along with configuration data:

```
…omitted for brevity…
objectsid                       : {S-1-5-21-1697496143-211471450-978293410-1217}
samaccounttype                  : MACHINE_ACCOUNT
objectguid                      : 94384c29-8ea7-404d-b575-9c45c3065709
useraccountcontrol              : WORKSTATION_TRUST_ACCOUNT
name                            : CORP-WIN-SERV-1
distinguishedname               : CN=CORP-WIN-SERV-
1,OU=Computers,OU=Corporate,DC=attackrange,DC=local
…omitted for brevity…
```

FIGURE 7 - Domain computer discovery

With a list of domain-joined machines and operating system information, the team used SharpView again to scan the network for available SMB network shares that were accessible from the `corp-win-serv-0 machine`:

```
…omitted for brevity…
Name                            : ADMIN$
ComputerName                    : corp-win-serv-1.attackrange.local
Name                            : C$
ComputerName                    : corp-win-serv-1.attackrange.local

Name                            : private
ComputerName                    : corp-win-serv-4.attackrange.local
…omitted for brevity…
```

FIGURE 8 - SMB share enumeration

The network shares revealed that the account CORPADMIN had administrative rights to the five machines in the corporate environment, along with 15 shares named public and one named private on the corp-win-serv-4 machine. The team used PowerShell to collect all of the files from the identified shares and staged them into the previously created C:\ProgramData\TempData. At this point, the team attempted to proxy psexec traffic in attempts to move laterally to corp-win-serv-1, which was detected by Windows Defender and resulted in losing initial Sliver agent connectivity. The team relaunched the agent using RDP and continued to manually examine the files collected from the network shares. This revealed a file called note.txt, which had been collected from corp-win-serv-1, containing cleartext credentials for the account STGADMIN.

With the newly acquired credentials, the team executed cmd.exe on corp-win-serv-0 via RDP using the runas command in order to conduct additional domain discovery from the perspective of the STGADMIN user. The team reused SharpView and the previous share discovery in attempts to identify additional assets exposed to the new user:

```
…omitted for brevity…
Name                               : private
ComputerName                       : stg-win-serv-0.attackrange.local
…omitted for brevity…
```

FIGURE 9 - SMB share enumeration on staging environment

Analyzing the results led the team to identify the previously inaccessible share located at \\stg-win-serv-0\private. The team proxied SMB traffic through the Sliver agent, using smbclient and proxychains with the newly discovered credentials to access the share and download the single note.txt file contained in the folder. This note.txt file contained an additional set of credentials for the user PRDADMIN.

Continuing to follow the methodology, the team reverted back to the network share discovery phase. The team again executed runas with the context of PRDADMIN on the corp-win-serv-0 machine over RDP and executed SharpView to discover any new available shares:

```
…omitted for brevity…
Name                               : private
ComputerName                       : prod-win-serv-3.attackrange.local
…omitted for brevity…
```

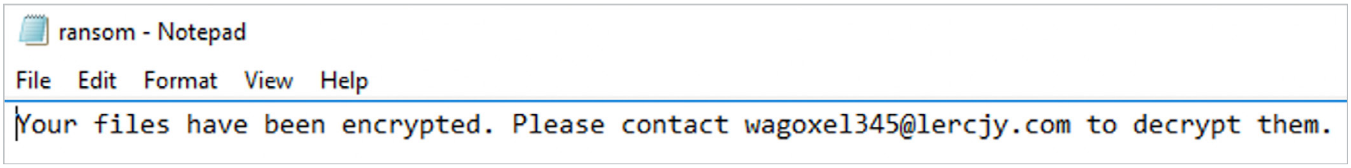
FIGURE 10 - Subdomain discovery on production environment

The team repeated the process of proxying SMB traffic in order to assess the contents of the newly discovered private share and discovered another note.txt file containing the cleartext credentials to the domain administrator account.

With the domain administrator credentials, the team logged into the win-dc server over RDP and dropped an additional Sliver agent on the domain controller. After executing and receiving a valid callback from the new Sliver agent, the team executed an additional domain share discovery to identify any additional assets available to the team before staging and exfiltrating the data.

The red team also tested additional payloads to validate lateral movement methodologies for future attacks in similar environments. Lateral movement TTPs tested included psexec and wmiexec. With successful compromise of the ATTACKRANGE\Administrator account, the team had successfully escalated privileges within the environment to a state where they could deploy ransomware targeting the previously discovered network shares.

Before deploying ransomware, the team exfiltrated the staged network files to a MEGA Cloud account using `rclone`. With the files successfully exfiltrated from the network, the team deployed ransomware that encrypted all network share files and then replaced the contents with the following text:



```
Your files have been encrypted. Please contact wagoxel345@lercjy.com to decrypt them.
```

FIGURE 11 - Ransomware deployed

With the ransomware successfully deployed, this marked the end of the control test scenario after an execution time of two hours and 28 minutes.

SCENARIO 2: DETECTION AND RESPONSE

Setup

In this scenario, the Illumio VEN agent was installed on all Windows systems, and labels were applied to the workloads in the Illumio Policy Compute Engine (PCE) interface. However, the agents were paired with a profile set to the Visibility Only (passive) mode, which only logged observed network traffic without blocking any of it.

The goal was to leverage the telemetry generated by Sysmon, Aurora, and the Illumio VEN agents to detect a live attack and implement a containment policy using Illumio to limit damage.

Test Results

The red team started the scenario in a similar manner to the previous run by creating an RDP session to `corp-win-serv-0` using previously compromised credentials. Following the established methodology, the team conducted local host discovery using the RDP session and a Windows command line to identify locally running processes, network configuration, and cached details of machines on the network.

After identifying additional machines on the network, the team used RDP to pivot to `corp-win-serv-4` using the `CORPADMIN` account. This activity was an attempt to have the `corp-win-serv-0` session remain active in the event of detected activities on `corp-win-serv-4`.

With the new RDP session on `corp-win-serv-4`, the team manually browsed the local filesystem and found the `join_domain.ps1` PowerShell script located in the `C:\ directory`. The PowerShell script contained details on the logical separation of the network into staging, production, and corporate machines. The team proceeded to upload a staged Sliver agent DLL to `C:\ProgramData\` using the RDP session on `corp-win-serv-4`. The team used a signed binary proxy execution technique to inject the payload using `rundll32.exe`.

After receiving the agent's callback, the team used defensive evasion techniques to unhook API calls and sideloaded a secondary Pneuma post-exploitation agent as a fallback. The team continued with local host discovery by enumerating the list of running processes. Shortly after the injection of the secondary agent and the process exploration, both agents on `corp-win-serv-4` lost connection to the command-and-control (C2) host.

The red team attempted to move laterally to `corp-win-serv-3` from `corp-win-serv-0` to remain on the network. The team uploaded another Pneuma agent to `corp-win-serv-3` and promptly created a staging directory in `C:\ProgramData\TempData`. The red team then continued the methodology with network and domain discovery by querying domain controller information and attempting to find passwords in GPP files.

After the red team enumerated GPP network shares, the blue team successfully blocked the identified C2 hosts based on alerts of the previous activities. The red team lost all active C2 and RDP sessions into the network, which marked the end of the scenario, with an execution time of 38 minutes. This demonstrated a strong response to common ransomware TTPs, especially regarding Illumio's capabilities in regard to network segmentation and stopping lateral movement.

SCENARIO 3: PRECONFIGURED STATIC PROTECTION

Setup

Similar to scenario 2, all Windows systems had the VEN agent installed, and labels were applied to all workloads. However, this time the VEN agent was deployed in Full Enforcement mode with a basic segmentation policy that would block known-bad ransomware strains, emulated here by the IP addresses, domain names, and network ports used in the previous attempts as a baseline.

The goal was to ensure that a list of known malware could be blocked by a segmentation policy. In this scenario, the attacker did not adjust payloads and just executed the ones from the previous scenarios.

Test Results

The third scenario focused on testing Illumio's enforcement mode configuration, with the C2 mechanisms from the previous scenario added to a deny-list. The same TTPs and methodology from the previous scenario were used in order to keep this consistent with blocked payloads and C2 mechanisms.

The red team's initial attempts to connect to the `corp-win-serv-1` machine using RDP with the `CORPADMIN` credentials failed due to existing network restrictions. To allow the initial RDP connection into the lab, the blue team reconfigured the network access controls to allow RDP from one host in order to demonstrate Illumio's other blocking capabilities.

The team proceeded to create an RDP session to `corp-win-serv-1` and opened a Windows command-line prompt. The team conducted basic network and domain discovery using the built-in Windows net command to retrieve a list of domain administrators and domain controllers in the network. The team also enumerated the local machine by discovering running processes and network connections with `netstat`.

Next, the team attempted to download secondary payloads using PowerShell, but subsequent execution attempts were blocked by Microsoft Defender. The team then uploaded a Sliver agent using the RDP connection and attempted to execute it using the `rundll32.exe` technique from the previous scenario. This activity was also blocked by Microsoft Defender, which generated alerts for the blue team.

The red team continued attempts to conduct network discovery by uploading additional obfuscated payloads for enumeration of network shares. While the subsequent attempts at network share enumeration were successful, the results indicated that additional segmentation was enforced in the network. Before the red team could attempt to pivot to additional machines in the network, the blue team successfully severed all connections from the red team. Loss of communication to the network marked the end of this scenario after 24 minutes.

SCENARIO 4: FULL APPLICATION RING-FENCING

Setup

This scenario had the most complete segmentation policy. For each existing environment, the team created labels (API, Database, and Jump host) assigned to workloads with the following distribution:

- Two API workloads
- Two Database workloads
- One Jump host workload

The microsegmentation policy consisted of the following rules:

- Database workloads in one environment could not connect to other environments
- API workloads in one environment could not connect to other environments
- The Jump host workload from the Corporate environment could access every host in the Staging environment using RDP
- The Jump host workload from the Staging environment could access every host in the Production environment using RDP
- Every workload could communicate with the domain controller
- Every workload could access public SMB shares on all environments
- Every workload could communicate to the internet on the following ports:
  - 443/TCP
  - 80/TCP
  - 53/TCP
  - 53/UDP
  - 123/UDP
- RDP access was authorized from the internet to API workloads in the Corporate network as an entry point for the attacker

Like in previous scenarios, all telemetry from the EDR agent and Sysmon were forwarded to the centralized Splunk instance.

Test Results

The red team followed the same TTPs and methodology from the previous scenarios and started the round by connecting to `corp-win-serv-0` using the `CORPADMIN` account. The team then uploaded a Sliver agent to `C:\ProgramData\Amazon` and executed it. Microsoft Defender detected the initial payload, so the red team modified Defender to allow the binary and re-executed the payload.

After waiting several minutes for a C2 callback and fallback connection methods to execute, the red team still had no established session with the Sliver agent, indicating additional segmentation had been enforced. The red team followed methodology without a C2 agent and began local host enumeration using a Windows command prompt to enumerate running processes:

```
…omitted for brevity…
venAgentMonitor.exe          1508 Services                   0      5,804 K Unknown
NT AUTHORITY\SYSTEM                                    0:00:00 N/A
venAgentMgr.exe              1516 Services                   0     14,872 K Unknown
NT AUTHORITY\SYSTEM
MsMpEng.exe                     8 Services                   0    212,764 K Unknown
NT AUTHORITY\SYSTEM                                    0:02:33 N/A
splunkd.exe                  1992 Services                   0     65,776 K Unknown
NT AUTHORITY\SYSTEM
…omitted for brevity…
```

FIGURE 12 - Enumerating active processes

The team continued the discovery process and identified the password policies in place, along with local user accounts on the machine, before losing the RDP session due to blue team countermeasures. The loss of network access marked the end of the scenario, which concluded 10 minutes after it started.

Additional Testing

To emulate a more advanced threat actor, the red team performed more evasive actions during the execution of scenario 4 to keep a low profile and avoid detection. Making these significant changes to the testing methodology required the team to extract this specific test case in its own category, which is presented here.

SCENARIO 5: PRECONFIGURED STATIC PROTECTION WITH DYNAMIC UPDATES

Setup

This scenario had a very similar setup to scenario 3: A default enforcement boundary was created to block all workloads from communicating with known-bad C2 hosts. On top of this policy, dynamic updates could be made to attempt to contain and stop the attack as it was occurring.

Test Results

This scenario focused on testing Illumio's preconfigured static protection configurations with dynamic reactions from the blue team to attempt to stop lateral movement throughout the network. The red team updated their TTPs to include the rotation of C2 hosts, additional obfuscation of payloads, and the addition of a SOCKS proxy tool for tunneling of external traffic.

The red team initiated an RDP session to `corp-win-serv-0` using previously compromised `CORPADMIN` credentials. After gaining desktop access to the target, the team uploaded a Sliver agent to `C:\ProgramData\Amazon` and executed it. The team also uploaded an obfuscated copy of Ligolo, a reverse tunneling SOCKS proxy, in an attempt to maintain network access if the Sliver agent lost connection. After receiving a callback from the Ligolo client, the red team verified access by proxying traffic to `api.ipify.org`, which triggered an alert for the blue team:

```
Detects DNS queries for ip lookup services such as api.ipify.org not originating from a browser process.
```

FIGURE 13 - ipify request from non-browser

The red team continued with the methodology and began network and domain discovery by querying domain controller information, including a list of domain administrators for future targeting. The red team also discovered local network connections with `netstat`, which revealed connections to a Splunk server on the network.

In attempts to maintain a network foothold, the red team pivoted an RDP connection to `corp-win-serv-4`, then uploaded and executed a Sliver agent. With the new connection, the team resumed network discovery by using SharpView to enumerate all other domain connected machines:

```
…omitted for brevity…
dnshostname                          : win-dc.attackrange.local
dnshostname                          : stg-win-serv-1.attackrange.local
dnshostname                          : prod-win-serv-0.attackrange.local
…omitted for brevity…
```

FIGURE 14 - Domain computer enumeration

The team continued with SMB share discovery, which revealed available network shares including one named private on `corp-win-serv-4`. With the identified shares, the team created a staging directory in `C:\ProgramData\TempData` and uploaded a PowerShell script for staging data from the identified network shares. After failed attempts to move laterally through RDP to `corp-win-serv-1`, the team uploaded a copy of Rclone for exfiltrating the previously staged network share data. The team successfully exfiltrated the data to MEGA and reviewed it manually, which led to the identification of the `STGADMIN` credentials.

After the previous network enumeration activity, the team pulled cached ARP details to get the resolved IP addresses for Nmap scanning through the previously established SOCKS proxy:

```
Internet Address      Physical Address      Type
  10.0.1.1              02-c3-a3-e3-7b-e7     dynamic
  10.0.1.12             02-58-5b-df-88-e7     dynamic
  10.0.1.14             02-1d-a6-cf-ce-63     dynamic
  10.0.1.30             02-3e-59-b8-6f-61     dynamic
…omitted for brevity…
```

FIGURE 15 - Cached ARP hosts

The team attempted to extract cached credentials using Mimikatz on `corp-win-serv-0`, but the attempts failed due to not escalating to a high-integrity process. The team attempted to migrate the Sliver agent into `explorer.exe` for further defense evasion, which caused the agent to lose connection. After executing the Sliver agent a second time using the RDP session, the team used an obfuscated version of Rubeus to conduct Kerberoasting attacks against the domain controller. This resulted in the red team receiving three hashes for the same users from the control test.

The red team executed the Sliver agent again as `Administrator`, bypassing UAC with the RDP session in order to gain system-level privileges on `corp-win-serv-0`. The team executed `getprivs` to verify the new access had been successful. With the high-integrity process, the team attempted to use Mimikatz again, which was successful but did not result in any higher privileges.

After the red team extracted the LSASS process from Mimikatz, the blue team successfully updated segmentation rules, which resulted in the red team losing all access to the environment. The blue team activities marked the end of the preconfigured static emulation after one hour and 25 minutes.

Appendix A — List of Tactics, Techniques, and Procedures

All TTPs used by the Bishop Fox red team during this assessment are listed below.

ATT&CK ID	NAME	TACTIC
T1003.001	OS Credential Dumping: LSASS Memory	Credential Access
T1005	Data from Local System	Collection
T1008	Fallback Channels	Command and Control
T1012	Query Registry	Discovery
T1018	Remote System Discovery	Discovery
T1021.001	Remote Services: Remote Desktop Protocol	Lateral Movement
T1021.002	Remote Services: SMB/Windows Admin Shares	Lateral Movement
T1021.006	Remote Services: Windows Remote Management	Lateral Movement
T1033	System Owner/User Discovery	Discovery
T1047	Windows Management Instrumentation	Execution
T1049	System Network Connections Discovery	Discovery
T1057	Process Discovery	Discovery
T1059.001	Command and Scripting Interpreter: PowerShell	Execution
T1069	Permission Groups Discovery	Discovery
T1074	Data Staged	Collection
T1078	Valid Accounts	Initial Access, Privilege Escalation, Defense Evasion
T1083	File and Directory Discovery	Discovery
T1106	Native API	Execution
T1115	Clipboard Data	Collection
T1134	Access Token Manipulation	Privilege Escalation
T1135	Network Share Discovery	Discovery

ATT&CK ID	NAME	TACTIC
T1218.011	Signed Binary Proxy Execution: Rundll32	Defense Evasion
T1482	Domain Trust Discovery	Discovery
T1486	Data Encrypted for Impact	Impact
T1518.001	Software Discovery: Security Software Discovery	Discovery
T1552.006	Unsecured Credentials: Group Policy Preferences	Credential Access
T1558.003	Steal or Forge Kerberos Tickets: Kerberoasting	Credential Access
T1558.004	Steal or Forge Kerberos Tickets: AS-REP Roasting	Credential Access
T1560.001	Archive Collected Data: Archive via Utility	Collection
T1562.001	Impair Defenses: Disable or Modify Tools	Defense Evasion
T1563.002	Remote Service Session Hijacking: RDP Hijacking	Lateral Movement
T1567.002	Exfiltration Over Web Service: Exfiltration to Cloud Storage	Exfiltration
T1570	Lateral Tool Transfer	Lateral Movement
T1572	Protocol Tunneling	Command and Control
T1573	Encrypted Channel	Command and Control
T1620	Reflective Code Loading	Defense Evasion

The following sections list the TTPs that the red team was able to attempt during each round of testing. **Note that as the Zero Trust Segmentation policies become increasingly strict, the red team is able to attempt fewer TTPs.**

SCENARIO 1

ATT&CK ID	NAME	TACTIC	SUCCESSFUL
T1005	Data from Local System	Collection	Yes
T1012	Query Registry	Discovery	Yes
T1018	Remote System Discovery	Discovery	Yes
T1021.001	Remote Services: Remote Desktop Protocol	Lateral Movement	Yes
T1021.002	Remote Services: SMB/Windows Admin Shares	Lateral Movement	Yes
T1033	System Owner/User Discovery	Discovery	Yes
T1047	Windows Management Instrumentation	Execution	Yes
T1049	System Network Connections Discovery	Discovery	Yes
T1057	Process Discovery	Discovery	Yes
T1059.001	Command and Scripting Interpreter: PowerShell	Execution	Yes
T1069	Permission Groups Discovery	Discovery	Yes
T1074	Data Staged	Collection	Yes
T1078	Valid Accounts	Initial Access, Privilege Escalation, Defense Evasion	Yes
T1083	File and Directory Discovery	Discovery	Yes
T1106	Native API	Execution	Yes
T1135	Network Share Discovery	Discovery	Yes
T1482	Domain Trust Discovery	Discovery	Yes
T1486	Data Encrypted for Impact	Impact	Yes
T1518.001	Software Discovery: Security Software Discovery	Discovery	Yes
T1552.006	Unsecured Credentials: Group Policy Preferences	Credential Access	Yes
T1558.003	Steal or Forge Kerberos Tickets: Kerberoasting	Credential Access	Yes
T1558.004	Steal or Forge Kerberos Tickets: AS-REP Roasting	Credential Access	Yes
T1567.002	Exfiltration Over Web Service: Exfiltration to Cloud Storage	Exfiltration	Yes
T1570	Lateral Tool Transfer	Lateral Movement	Yes
T1572	Protocol Tunneling	Command and Control	Yes
T1573	Encrypted Channel	Command and Control	Yes

SCENARIO 2

ATT&CK ID	NAME	TACTIC	SUCCESSFUL
T1005	Data from Local System	Collection	Yes
T1018	Remote System Discovery	Discovery	Yes
T1021.001	Remote Services: Remote Desktop Protocol	Lateral Movement	Yes
T1033	System Owner/User Discovery	Discovery	Yes
T1057	Process Discovery	Discovery	Yes
T1074	Data Staged	Collection	Yes
T1078	Valid Accounts	Initial Access, Privilege Escalation, Defense Evasion	Yes
T1135	Network Share Discovery	Discovery	Yes
T1218.011	Signed Binary Proxy Execution: Rundll32	Defense Evasion	Yes
T1552.006	Unsecured Credentials: Group Policy Preferences	Credential Access	Yes
T1570	Lateral Tool Transfer	Lateral Movement	Yes
T1573	Encrypted Channel	Command and Control	Yes
T1620	Reflective Code Loading	Defense Evasion	No

SCENARIO 3

ATT&CK ID	NAME	TACTIC	SUCCESSFUL
T1005	Data from Local System	Collection	Yes
T1018	Remote System Discovery	Discovery	Yes
T1021.001	Remote Services: Remote Desktop Protocol	Lateral Movement	Yes
T1033	System Owner/User Discovery	Discovery	Yes
T1059.001	Command and Scripting Interpreter: PowerShell	Execution	Yes
T1078	Valid Accounts	Initial Access, Privilege Escalation, Defense Evasion	Yes
T1135	Network Share Discovery	Discovery	Yes
T1218.011	Signed Binary Proxy Execution: Rundll32	Defense Evasion	Yes
T1573	Encrypted Channel	Command and Control	Yes

SCENARIO 4

ATT&CK ID	NAME	TACTIC	SUCCESSFUL
T1008	Fallback Channels	Command and Control	No
T1018	Remote System Discovery	Discovery	Yes
T1021.001	Remote Services: Remote Desktop Protocol	Lateral Movement	Yes
T1033	System Owner/User Discovery	Discovery	Yes
T1057	Process Discovery	Discovery	Yes
T1078	Valid Accounts	Initial Access, Privilege Escalation, Defense Evasion	Yes
T1562.001	Impair Defenses: Disable or Modify Tools	Defense Evasion	Yes
T1573	Encrypted Channel	Command and Control	No

ADDITIONAL TESTING: SCENARIO 5

ATT&CK ID	NAME	TACTIC	SUCCESSFUL
T1003.001	OS Credential Dumping: LSASS Memory	Credential Access	Yes
T1005	Data from Local System	Collection	Yes
T1008	Fallback Channels	Command and Control	Yes
T1018	Remote System Discovery	Discovery	Yes
T1021.001	Remote Services: Remote Desktop Protocol	Lateral Movement	Yes
T1021.002	Remote Services: SMB/Windows Admin Shares	Lateral Movement	Yes
T1033	System Owner/User Discovery	Discovery	Yes
T1047	Windows Management Instrumentation	Execution	No
T1049	System Network Connections Discovery	Discovery	Yes
T1057	Process Discovery	Discovery	Yes
T1059.001	Command and Scripting Interpreter: PowerShell	Execution	Yes
T1069	Permission Groups Discovery	Discovery	Yes
T1074	Data Staged	Collection	Yes
T1078	Valid Accounts	Initial Access, Privilege Escalation, Defense Evasion	Yes

ATT&CK ID	NAME	TACTIC	SUCCESSFUL
T1106	Native API	Execution	Yes
T1134	Access Token Manipulation	Privilege Escalation	Yes
T1135	Network Share Discovery	Discovery	Yes
T1518.001	Software Discovery: Security Software Discovery	Discovery	Yes
T1558.003	Steal or Forge Kerberos Tickets: Kerberoasting	Credential Access	Yes
T1558.004	Steal or Forge Kerberos Tickets: AS-REP Roasting	Credential Access	Yes
T1567.002	Exfiltration Over Web Service: Exfiltration to Cloud Storage	Exfiltration	Yes
T1570	Lateral Tool Transfer	Lateral Movement	Yes
T1572	Protocol Tunneling	Command and Control	Yes
T1573	Encrypted Channel	Command and Control	Yes
T1620	Reflective Code Loading	Defense Evasion	No

About Illumio

Illumio, the Zero Trust Segmentation company, prevents breaches from spreading and turning into cyber disasters. Illumio protects critical applications and valuable digital assets with proven segmentation technology purpose-built for the Zero Trust security model. Illumio ransomware mitigation and segmentation solutions see risk, isolate attacks, and secure data across cloud-native apps, hybrid and multi-clouds, data centers, and endpoints, enabling the world's leading organizations to strengthen their cyber resiliency and reduce risk. For more information, visit **illumio.com**.

About Bishop Fox

Bishop Fox is the leading authority in offensive security, providing solutions ranging from continuous penetration testing, red teaming, and attack surface management to product, cloud, and application security assessments. We've worked with more than 25% of the Fortune 100, half of the Fortune 10, eight of the top 10 global technology companies, and all of the top global media companies to improve their security. Our Cosmos platform was named **Best Emerging Technology in the 2021 SC Media Awards** and our offerings are consistently ranked as "world class" in customer experience surveys. We've been actively contributing to and supporting the security community for almost two decades and have published more than 16 open-source tools and 50 security advisories in the last five years. Learn more at **bishopfox.com** or follow us on **Twitter**.

8240 S. Kyrene Rd. • Tempe, AZ 85284
480.621.8967
hello@bishopfox.com • bishopfox.com