AvosLocker – Modern Linux Ransomware Threats

By Threat Analysis Unit

Published: 2022-02-25 · Archived: 2026-04-05 18:11:43 UTC

This article was written by Sudhir Devkar

Summary

AvosLocker Ransomware is a recent ransomware with the capability to encrypt Linux systems. AvosLocker seems to be targeting the VMware ESXi virtual machines and Virtual Machine File System (VMFS) files. By targeting VMs, AvosLocker takes advantage of faster and easier encryption of multiple servers with a single command.

Behavioral Summary

On execution, AvosLocker on Linux systems shows usage instructions to the user to run commands with parameters, as shown in Figure 1. These parameters control aspects like the number of threads to be created for encryption and the path of the directory which will get encrypted.

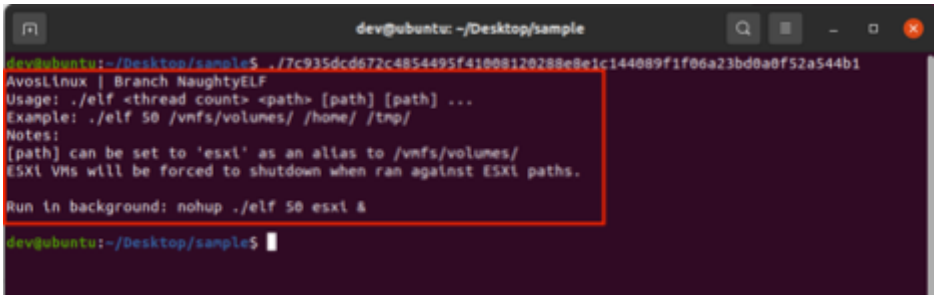

Figure 1: Command Line Usage guide

After providing the parameters, before encryption it drops ransom note files to folders specified on the command line with the name “README_FOR_RESTORE”. In the ransom note AvosLocker asks the user to download the Tor browser and to visit the given Tor onion link. There is no specific ransom amount demanded in the ransom note; it instructs to provide the ID mentioned at the end in the ransom note to get pricing details.

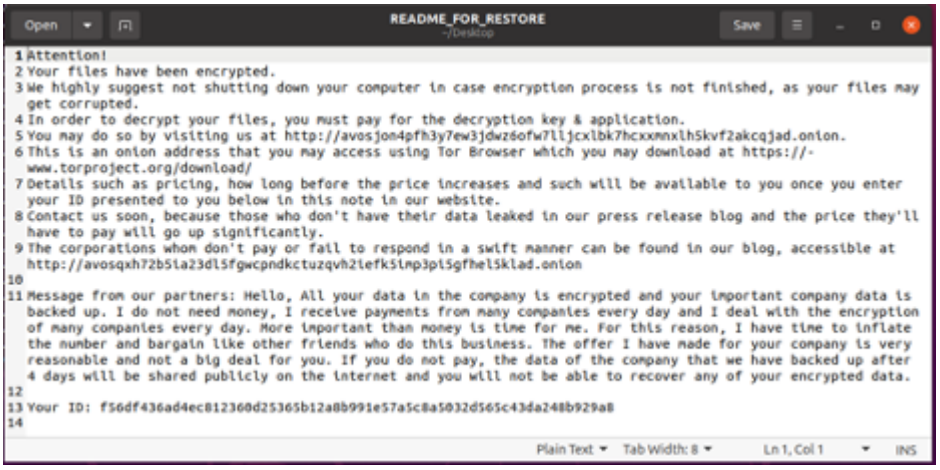

Figure 2: Ransom Note

While looking into code, the malware checks if the command line parameter contains “esxi” and “vmfs”. If so, AvosLocker checks for VMware Elastic Sky X Integrated (ESXi) and Virtual Machine File System (VMFS), respectively, and tries to force their shutdown if they are running.

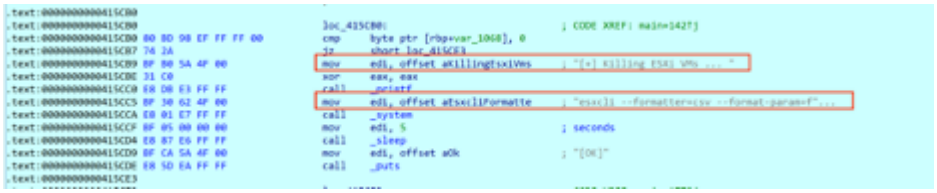

Figure 3: Code snippet to kill ESXi

Command used to kill ESXi and VMFS services:

esxcli –formatter=csv –format-param=fields==”WorldID,DisplayName” vm process list | tail -n +2 | awk -F $’,’ ‘{system(“esxcli vm process kill –type=force –world-id=” $1)}’

Further, it creates a given number of threads with mutex lock/unlock to synchronise operation to prevent encryption process overlap.

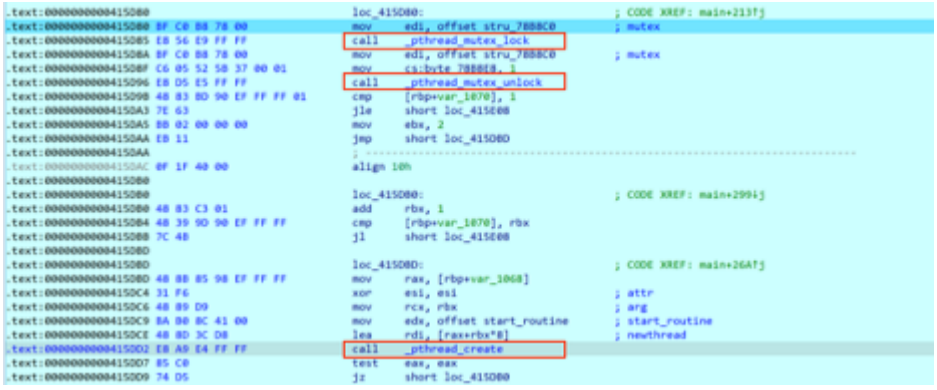

Figure 4: Create Thread and Mutex lock/Unlock

On further execution AvosLocker creates the number of threads provided by command line parameters and starts encrypting files from a given path. Analysis showed that each file was encrypted with a unique encryption key. During encryption AvosLocker checks file size if greater than ~12MB. If so, then the data will be encrypted in ~1Mb blocks. Once encryption completes, the malware stores the encryption key, with base64 encoding, at the end of each encrypted file. The ransomware then renames the encrypted file with extension ".avoslinux", as shown in Figure:5

```
v10 = sizeOffile;
if ( sizeOffile > 12287999 )                    // file size check ~12MB
{
  while ( 1 )
  {
    BytesRead = fread(malloc_h, 1uLL, 1024000uLL, fopen_h2);// Read file ~1MB
    v12 = v10 - BytesRead;
    fseek(fopen_h2, -BytesRead, 1);
    EncryptionCall(v28, malloc_h, malloc_h, BytesRead);// Encryption function
    fwrite(malloc_h, 1uLL, BytesRead, fopen_h2);
    if ( BytesRead <= 1023999 )                 // Read bytes check if less than 1MB
      break;
    if ( v12 <= 10244095 )                      // remaining bytes in file less than ~10MB
      goto LABEL_25;
    v10 = v12 - 10240000;
    fseek(fopen_h2, 10240000LL, 1);
  }
}
else
{
  BytesRead_1 = fread(malloc_h, 1uLL, 1024000uLL, fopen_h2);// Read file
  fseek(fopen_h2, 0LL, 0);
  EncryptionCall(v28, malloc_h, malloc_h, BytesRead_1);// Encryption function
  if ( sizeOffile <= 1023828 )                  // file size check ~1MB
  {
    Base64DataAppend(&malloc_h[BytesRead_1], 171LL, v27, 171LL);// appends 171 byte sized encoded base64 to encrypted data
    fwrite(malloc_h, 1uLL, BytesRead_1 + 171, fopen_h2);
    goto LABEL_12;
  }
  fwrite(malloc_h, 1uLL, BytesRead_1, fopen_h2);
LABEL_25:
  fseek(fopen_h2, 0LL, 2);
}
fwrite(v27, 1uLL, 171uLL, fopen_h2);            // appends 171 byte sized encoded base64 to encrypted data
LABEL_12:
free(malloc_h);
fclose(fopen_h2);
sub_41CAE0(&v29, a1, ".avoslinux");
v13 = v29[0];
rename(*a1, v29[0]);                            // Rename encrypted file with extension .avoslinux
std::string::~string(v29);
```

Figure 5: Encryption code flow

The encrypted files are appended with 171 bytes of base64 data. Analysis of code flow shows this to be the encryption key stored in base64 encoded, shown in Figure:6.

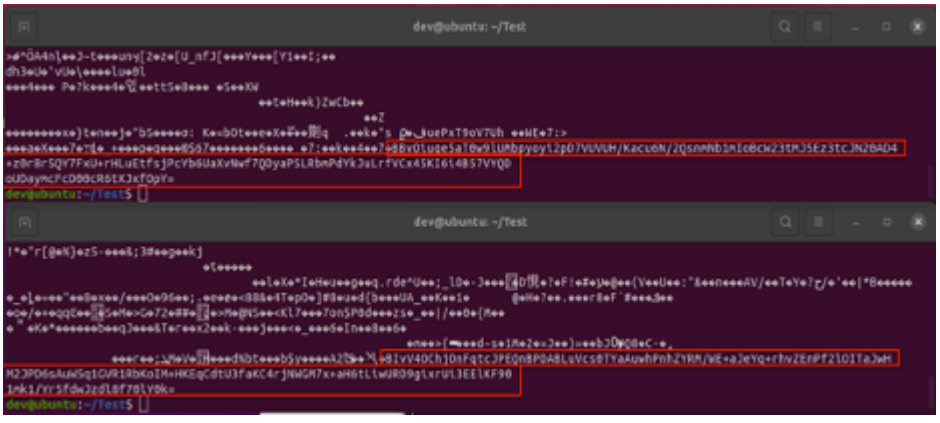

Figure 6: key appended in encrypted files

After encryption, AvosLocker appends the encrypted file name with the extension **.avoslinux.** (Figure:7)

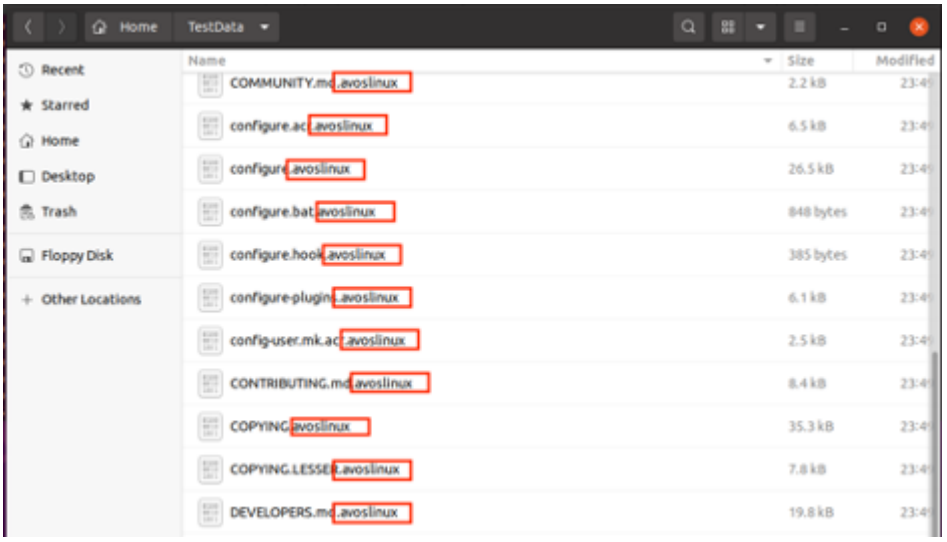

Figure 7: Encrypted files with .avoslinux extension

Yara

Rule :

rule AvosLocker {

meta:

description = “AvosLocker Ransomware”

author = “VMware Threat Research”

exemplar_hashes = “7c935dcd672c4854495f41008120288e8e1c144089f1f06a23bd0a0f52a544b1”

strings:

$s1 = “avoslinux” wide ascii nocase

$s2 = “README_FOR_RESTORE” wide ascii nocase

$s3 = “Killing ESXi VMs” wide ascii nocase

condition:

uint32(0) == 0x464C457F and filesize > 1MB and filesize < 3MB and

all of ($s*)

}

MITRE ATT&CK TIDs

TID	Tactic	Description

T1490	Impact	Inhibit System Recovery
T1489	Impact	Service Stop
T1486	Impact	Data Encrypted for Impact
T1082	Discovery	System Information Discovery
T1059	Execution	Command and Scripting Interpreter

Table 1: MITRE ATT&CK TIDs

Indicators of Compromise (IOCs)

Indicator	**Type**	**Context**
7c935dcd672c4854495f41008120288e8e1c144089f1f06a23bd0a0f52a544b1	SHA256	AvosLocker ELF
0cd7b6ea8857ce827180342a1c955e79c3336a6cf2000244e5cfd4279c5fc1b6	SHA256	AvosLocker ELF
10ab76cd6d6b50d26fde5fe54e8d80fceeb744de8dbafddff470939fac6a98c4	SHA256	AvosLocker ELF
e9a7b43acdddc3d2101995a2e2072381449054a7d8d381e6dc6ed64153c9c96a	SHA256	AvosLocker ELF
e737c901b80ad9ed2cd800fec7c2554178c8afab196fb55a0df36acda1324721	SHA256	AvosLocker ELF
cdca6936b880ab4559d3d96101e38f0cf58b87d07b0c7bf708d078c2bf209460	SHA256	AvosLocker decryptor ELF
05c63ce49129f768d31c4bdb62ef5fb53eb41b54	SHA1	AvosLocker ELF
6f110f251860a7f6757853181417e19c28841eb4	SHA1	AvosLocker ELF
9c8f5c136590a08a3103ba3e988073cfd5779519	SHA1	AvosLocker ELF
e8c26db068914df2083512ff8b24a2cc803ea498	SHA1	AvosLocker ELF
dab33aaf01322e88f79ffddcbc95d1ad9ad97374	SHA1	AvosLocker ELF

e60ef891027ac1dade9562f8b1de866186338da1	SHA1	AvosLocker decryptor ELF
e09183041930f37a38d0a776a63aa673	MD5	AvosLocker ELF
d3cafcd46dea26c39dec17ca132e5138	MD5	AvosLocker ELF
f659d1d15d2e0f3bd87379f8e88c6b42	MD5	AvosLocker ELF
afed45cd85a191fe3b2543e3ae6aa811	MD5	AvosLocker ELF
31f8eedc2d82f69ccc726e012416ce33	MD5	AvosLocker ELF
a39b4bea47c4d123f8195a3ffb638a1b	MD5	AvosLocker decryptor ELF

Table 2: Indicator of Compromise

Read more threat analysis insights.

Based on VMware's Threat Analysis Unit research, Exposing Malware in Linux-Based Multi-Cloud Environments offers a comprehensive look at malware threats targeting multi-cloud environments.
