Monti Ransomware Unleashes a New Encryptor for Linux

By Nathaniel Morales, Joshua Paul Ignacio (words)

Published: 2023-08-14 · Archived: 2026-04-05 22:45:56 UTC

Ransomware

The Monti ransomware collective has restarted their operations, focusing on institutions in the legal and governmental fields. Simultaneously, a new variant of Monti, based on the Linux platform, has surfaced, demonstrating notable differences from its previous Linux-based versions.

By: Nathaniel Morales, Joshua Paul Ignacio Aug 14, 2023 Read time: 5 min (1443 words)

Introduction

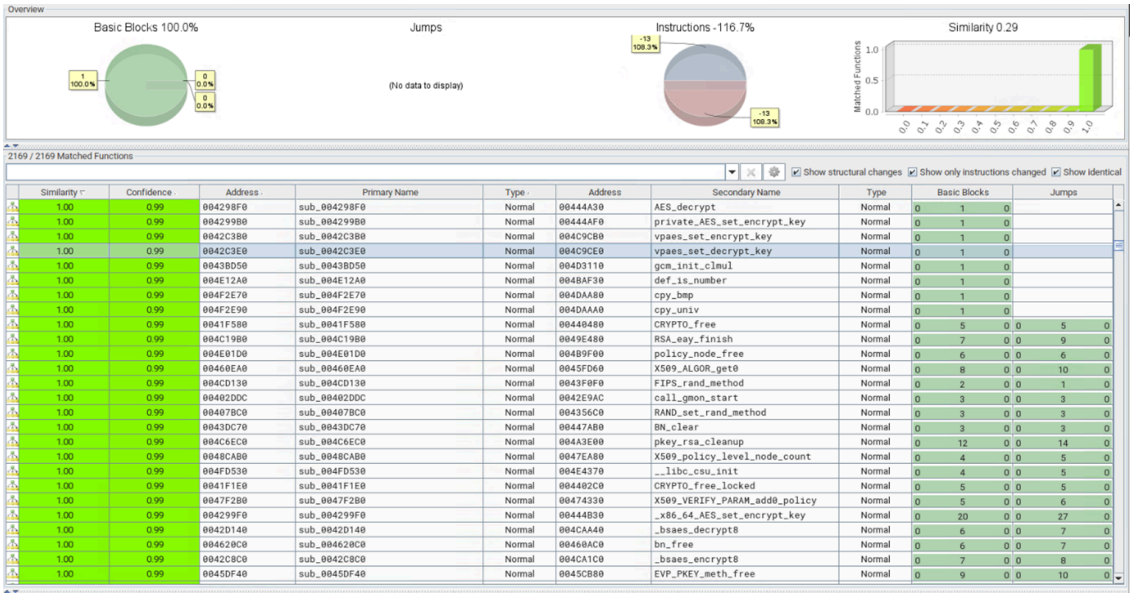
The Monti ransomware, which has both Windows and Linux-based variants, gained attention from cybersecurity organizations and researchers when it was first discovered in June 2022 because of its striking resemblance to the infamous Conti ransom ware — not just in name but also the tactics that the threat actors used. The group, operating under the moniker "Monti," has also deliberately emulated the widely recognized tactics, techniques, and procedures (TTPs) of the Conti team, incorporating a substantial number of their tools and even using Conti's leaked source code. Since its discovery, the Monti group has been continuously targeting companies, exposing them on their leak site.

Industry	Count
Legal	3
Financial services	2
Healthcare	2
Others	6

Table 1. The industries of the companies that appeared on the Monti ransomware leak site. Data is from March to August 2023.

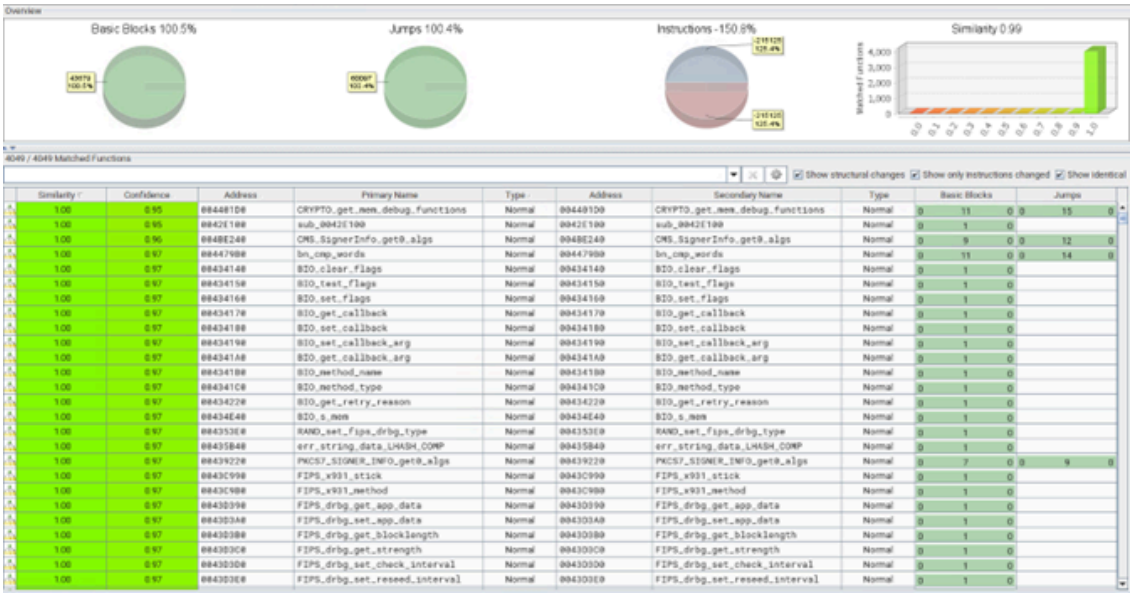
Following a two-month break from exposing victims on their leak site, the Monti ransomware group has resumed its malicious activities, this time targeting organizations within the legal and government sectors. Alongside this, a fresh Linux-based variant of Monti (Ransom.Linux.MONTI.THGOCBC) has emerged, displaying significant deviations from its other Linux-based predecessors. Unlike the earlier variant, which is primarily based on the leaked Conti source code, this new version employs a different encryptor with additional distinct behaviors. As of writing, only three security vendors that had the sample tagged it as malicious on VirusTotal

Comparing the new variant to the old one using BinDiff, we found that it only showed a similarity rate of 29% as opposed to the 99% similarity rate of the older variants and Conti.



[open on a new tab]

Figure 1. Comparison of the old and new Monti variants using BinDiff



[open on a new tab]

Figure 2. Comparison of the old Monti variant and Conti ransomware using Bindiff

Analysis

The new Linux variant accepts the following command line arguments, omitting some arguments from its older variant and adding the *--whitelist* parameter. The following table shows the added parameters in bold text while the removed parameters from the old variant are shown in italicized text.

Argument	Description
--help	Displays arguments usage
--path <string>	Path to be encrypted
--whitelist <string>	**List of VMs to be skipped**
--vmkill	Option to Kill virtual machine (VM)
--detach	Detach from terminal
--size	*removed*
--log	*removed*
--vmlist	*removed*

Table 2. The command line arguments accepted by the new variant

Compared to its predecessor, the current version also employs the *-type=soft* parameter to terminate virtual machines on the system (as opposed to the *--type=hard* parameter). The shift to *--type=soft* suggests that the threat actors behind Monti may have chosen this approach to minimize the risk of immediate detection while carrying out their activities.

```
if ( v33[0] != v33[1] )
{
  do
  {
    v11 = *v10;
    v12 = fork();
    if ( v12 == -1 )
    {
      perror("fork");
    }
    else
    {
      if ( !v12 )
      {
        execlp("esxcli", "esxcli", "vm", "process", 5234374LL, "--type=soft", "--world-id", v11, 0LL);
```

open on a new tab

Figure 3. Code snippet showing the -type=soft parameter used to terminate virtual machines

Monti's developers also tampered with the */etc/motd* and *index.html* files, replacing their contents with a ransom note announcing that the server has been successfully infiltrated. Note that MOTD (or Message of the Day) is a text message displayed when a user logs in to a Linux operating system.

Infection marker

One of the additions of this new variant is that it appends the bytes "MONTI" followed by an additional 256 bytes that is linked to the encryption key.

Before proceeding with its encryption routine, the ransomware will check specific conditions. First, it checks whether the file size is 261 bytes or below, which corresponds to the size of the infection marker it appends after

encryption. If this condition is met — indicating that the file is not encrypted given that its size is smaller than the appended infection marker — the ransomware proceeds with the infection process.

If the initial condition is not met, Monti will then check the last 261 bytes of the file to verify the presence of the string "MONTI." If this string is detected, the file will be skipped, signifying that it has already been encrypted. However, if the string is not found, the malware will proceed with the encryption process for the file.

```
  lseek(v4, -261LL, 2);
  v5 = old;
  v6 = "[%s] Error reading file meta before crypt.\n";
  if ( read(v4, &buf, 5uLL) == -1 )
  {
ERROR_LABEL:
    LOGGING_4058D0(v6, v5);
    LOWORD(v32) = 2;
    fcntl(v4, 7, &v32);
    close(v4);
    return 0;
  }
  lseek(v4, 0LL, 0);
  if ( buf == 0x544E4F4D && BYTE4(buf) == 0x49 )// 0x544E4F4D => "TNOM" 0x49 => "I"
  {
    v5 = old;
    v6 = "[%s] File already encrypted.\n";
    goto ERROR_LABEL;
  }
```

open on a new tab

Figure 7. Code snippet to check for the presence of the “MONTI” string via the last 261 bytes of the file to be encrypted

Checking file sizes and Intermittent encryption

Based on our analysis, the new ransomware variant employed AES-256-CTR encryption using *evp_enc* from the OpenSSL library instead of Salsa20, which is implemented by the old variant

We also discovered that the sample we analyzed employs various encryption methods for files. Unlike the previous variant, which utilized a *--size* argument to determine the percentage of the file to be encrypted, this new variant solely relies on the file size for its encryption process. In this section, we break down the different ways that the Monti ransomware determines the size of the file to be encrypted.

```
v7 = sub_410C00();
sub_40CC10(v25, v7, 0LL, &v36, &v38);
```

open on a new tab

Figure 8. Function containing the initialization of Cipher value stored in v7

```
        if ( v21 )
        {
          v22 = v21(a1, 0LL, 0LL, 0LL, v18);
          if ( v22 == -1 )
          {
            sub_40A260(6u, 124, 133, "evp_enc.c", 632);
          }
          else if ( v22 )
          {
            v11 = *a1;
            goto LABEL_24;
          }
        }
        else
        {
          sub_40A260(6u, 124, 132, "evp_enc.c", 626);
        }
        LODWORD(v18) = 195;
LABEL_54:
        sub_40A260(6u, 123, 134, "evp_enc.c", v18);
        return 0LL;
      }
LABEL_40:
      v27 = sub_41F220(v19, "evp_enc.c", 178LL);
      a1[15] = v27;
      if ( !v27 )
```

[open on a new tab]

Figure 9. Function 40CC10 containing evp_enc.c from the OpenSSL library

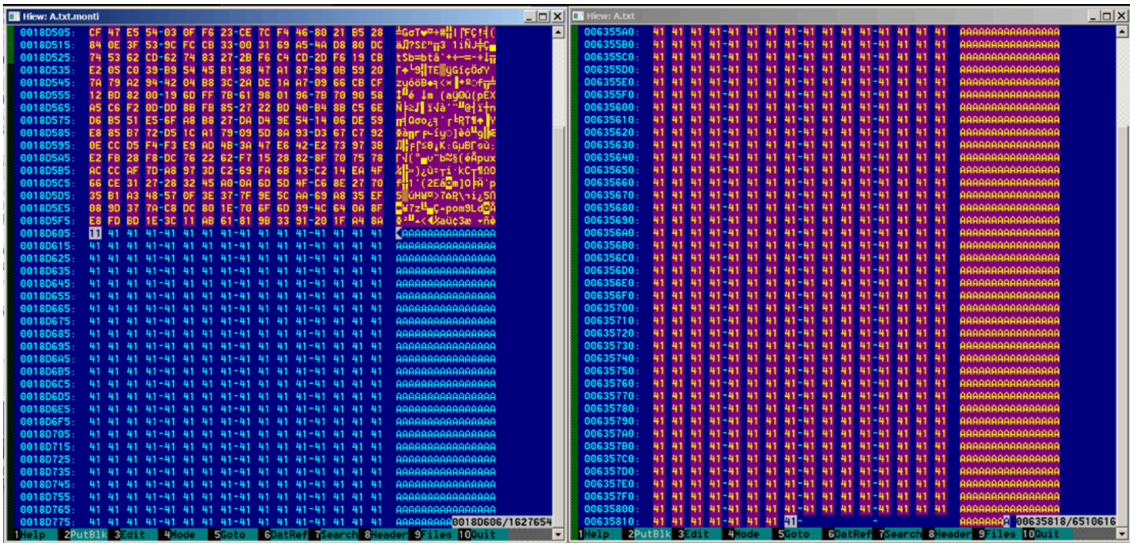
```
st_size = stat_buf.st_size;
if ( stat_buf.st_size > 0xFFFFF )          // if file size is greater than 1,048,575 bytes
{
  v10 = 0x100000LL;                        // set v10 to 100,000 bytes
  if ( stat_buf.st_size > 0x3FFFFF )       // if file size is greater than 4,194,303
    v10 = stat_buf.st_size >> 2;           // set v10 to filesize shift right 2
  st_size = v10;
}
```

[open on a new tab]

Figure 11. Checking file size and determining the size to be encrypted

The sample will only encrypt the first 100,000 (0xFFFFF) bytes of the file and append its infection marker at the end of the file if the file size is larger than 1.048MB but smaller than 4.19MB.

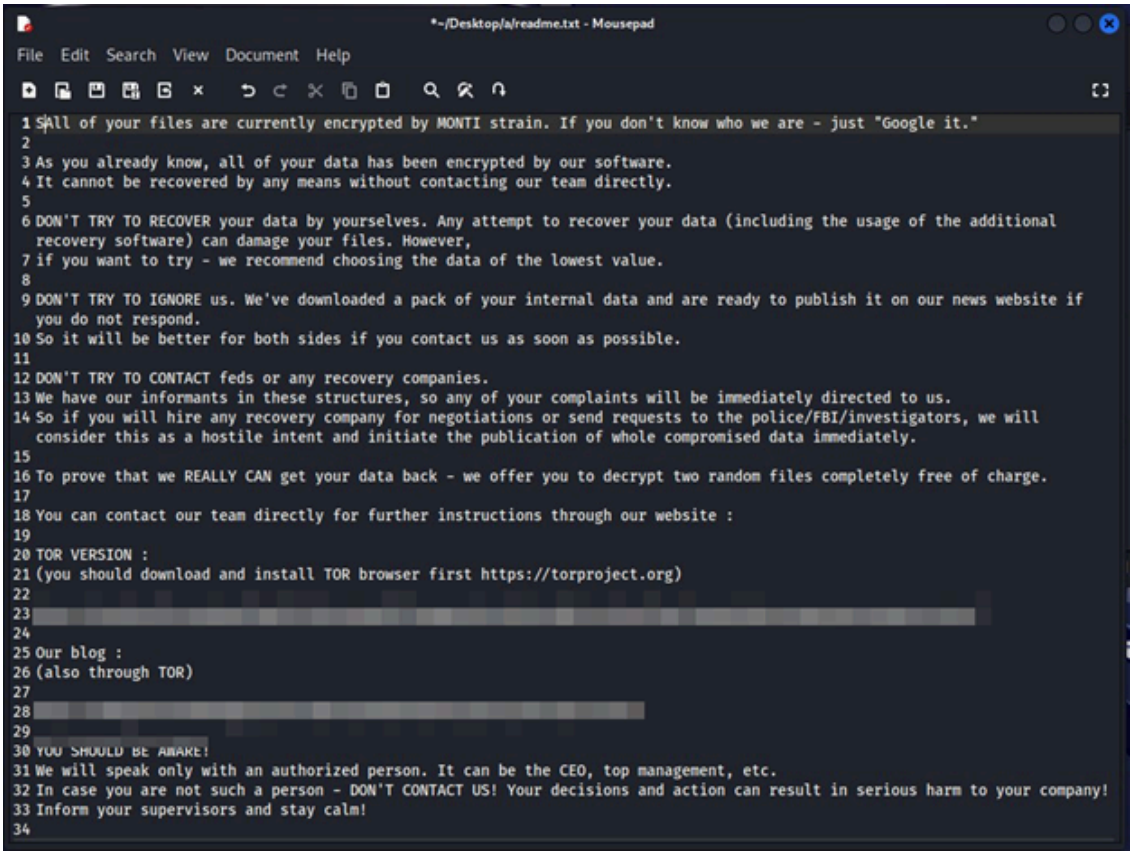
If the file size exceeds 4.19MB, it employs a Shift Right operation to calculate the total size of the file to be encrypted (which depends on the actual file size). Meanwhile, files with a size smaller than 1.048MB will have all their content encrypted.



open on a new tab

Figure 13. Encrypted file (left) vs original file (right). Using 0x635818(total size), Shift Right 2 is equivalent to 0x18D606 (bytes to be encrypted)

As with previous variants, the new version appends the *.monti* file extension to the encrypted files and drops its ransom note *readme.txt* to every directory.



open on a new tab

Figure 14. Appending the .monti suffix to encrypted files (top) and the ransom note

While analyzing the samples, we discovered a decryption code that suggests the threat actor was testing its functionality. It seems that they forgot to remove this code when deploying the sample. However, the decryption code is currently ineffective since it requires a private key known only to the malware author and has no connection to the malware routine. Therefore, it will not be executed by the program.

Conclusion

It's likely that the threat actors behind Monti still employed parts of the Conti source code as the base for the new variant, as evidenced by some similar functions, but implemented significant changes to the code — especially to the encryption algorithm. Furthermore, by altering the code, Monti's operators are enhancing its ability to evade detection, making their malicious activities even more challenging to identify and mitigate.

It is advisable for organizations to adopt effective defense strategies that include protocols for safeguarding data and the establishment of procedures for backup and recovery to protect their systems from ransomware attacks. These measures ensure the security of data and its potential restoration even in the event of encryption or deletion.

We suggest the subsequent security measures to protect important data:

- Implement multifactor authentication (MFA) to impede attackers from progressing horizontally within a network and gaining access to sensitive data.
- Adhere to the 3-2-1 guideline when generating backups for crucial files. This guideline entails creating three backup copies in two distinct file formats, with one copy stored at a separate location. This approach ensures redundancy and minimizes the possibility of data loss.

Trend Micro Solutions

A multilayered approach to security can help organizations guard possible entry points into their system such as endpoints, emails, web, and networks. The following security technologies can detect malicious components and suspicious behavior to help protect organizations from ransomware:

Trend Vision One™products provides multilayered protection and behavior detection, which helps block questionable behavior and tools early on before ransomware can do irreversible damage to the system.

Trend Cloud One™ – Workload Securityproducts protects systems against both known and unknown threats that exploit vulnerabilities. This protection is made possible through techniques such as virtual patching and machine learning.

Trend Micro™ Deep Discovery™ Email Inspectorproducts employs custom sandboxing and advanced analysis techniques to effectively block malicious emails, including phishing emails that can serve as entry points for ransomware.

Trend Micro Apex One™products offers next-level automated threat detection and response against advanced concerns such as fileless threats and ransomware, ensuring the protection of endpoints.

Additional Insights by Byron Gelera and Bren Matthew Ebriega

Indicators of Compromise

SHA1	Detection
f1c0054bc76e8753d4331a881cdf9156dd8b812a	Ransom.Linux.MONTI.THGOCBC
a0c9dd3f3e3d0e2cd5d1da06b3aac019cdbc74ef	Ransom.Linux.MONTI.THGADBC

- hxxp://monti5o7lvyrpyk26lqofnfvajtyqruwatlfaazgm3zskt3xiktudwid[.]onion
- hxxp://mblogci3rudehaagbryjznltdp33ojwzkq6hn2pckvjq33rycmzczpid[.]onion

Tags
