Part 1: Analysing MedusaLocker ransomware

By By Theta

Archived: 2026-04-05 16:25:22 UTC

Ransomware is a sad fact of life in 2020. While the “big game hunting” actors (Maze, REvil et al) get the attention of the media and industry by claiming high profile scalps, far less attention is given to the victim and attacker from smaller origins. There are a host of actors (often under-researched or reported) engaged in ransomware operations. Most of them could be classified as opportunistic rather than persistent.

In this post, we focus on pre-impact operations as a complete intrusion. These artefacts would normally be heavily degraded or destroyed by the ransomware, or by the need for a business to recover (often wiping evidence). You’ll see how we found and analysed a host that failed to encrypt correctly – this host also turned out to be the initial entry point and staging post for the operators. As we go through our analysis, we’ll point out which part of the kill chain the attacker is up to and where this maps to the MITRE ATT&CK framework.

It’s important to note that these adversaries weren’t particularly stealthy or advanced. They doubled up on a lot of tooling and functionality, yet they were able to completely overmatch the defenses presented to them. They achieved their goals by having enough effectiveness.

The ramifications of the attack

As a result of this attack, the organisation experienced the following:

Encryption of:

- 75% of end-user compute devices (these devices wouldn’t boot at all – we’re unsure if this was deliberate).
- 100%* of servers (however, the server that was the point of entry and staging post failed to encrypt completely, leaving parts of its system drive available for analysis).
- Mapped drives, shared drives, Dropbox, OneDrive (this organisation didn’t have enterprise licenses for its cloud file sharing solutions, making recovery more difficult).
- ERP System (over 500 hours to rebuild and move to a SaaS application). This resulted in all purchase/sales orders, inventories, ledgers, GST, shipping and financial records being lost.
- EDI services, which resulted in all automatic order placing and fulfilment being halted.
- Backups – variously encrypted or degraded. USB + network-attached storage schemes were all hit, forcing the rebuild of critical ERP and EDI solutions from much older backups - which added additional time and complexity.

Time from initial access to domain-wide encryption ~25hrs (16/06/2020 3:38 am to 17/06/2020 ~5 am)

Time to be able to trade again: 8 working days (and hundreds of recovery and rebuild hours behind the scenes plus manual trading with incremental improvements as data structures were recovered).

As we go through each stage of the attack, it's worth role playing this scenario in your own organisation to see how you would recover the loss of such data types. You can use the MITRE ATT&CK mapping to overlay your organisation's controls and your ability to detect and stop a similar attack.

This dataset comes from one intrusion without good instrumentation, and thus may not represent a total picture of the actor's TTP's. In the forensic analysis, bits were missing, and the pieces we found often made little sense. Still, by piecing together a timeline, we could be reasonably confident about the adversary activity that had happened.

Intrusion Analysis

Background

We received a phone call from the client on the morning of June 17th 2020. We'd supported their ERP and EDI systems in the past, but were not currently their IT outsource partner. They reported that all computers in their environment were not booting and that they thought they'd been attacked.

A search on Shodan showed one of their servers had RDP exposed, giving a possible source of entry (we were sadly correct). When our team arrived on site, we found their environment was completely shutdown; all of their workstations wouldn't boot, and all servers were powered off.

On the particular server that this analysis is from, we found its RAID array reporting it was in a degraded state (we're not sure if the ransomware did this) leaving us with some difficult choices when it came to triage and imaging. To our surprise, the ransomware failed to completely execute on this host. While all of the non-system drives (10+ TB of them) were completely encrypted, and its system drive was partially encrypted, we later saw that the task executing it had failed. Most of this analysis comes from imaging of the file system on that host.

Initial Access:

Initial Access was via RDP brute-forcing (T1110) to a Domain Controller that had RDP open to the internet (T1133 – External Remote Services). This (obviously dangerous) configuration was set up to enable remote access by the clients' main IT provider.

The initial logon via an admin account was recorded at 16/06/2020 3:38 am NZT from 185.202.1[.]19. Landing on an account with DA on a DC certainly made the job of the ransomware operator's life somewhat easier and shortened the kill chain. However, the assessment shows they would have been able to carry out their objectives given the low-security posture of the environment regardless. Logons were also observed from 213.7.208[.]69 & 5.2.224[.]56.

Execution:

Fragments of several interesting tool chains were recovered from the Server.

As the vector of entry was RDP, graphical tools were used (T1061) and recovered shellbags support this.

The intruder staged their tooling in a pre-existing folder - C:\SQLDB\.

Event logs record large amounts of PowerShell (T1086) activity being executed, although the system had PowerShell 2.0 installed, so no useful information was logged as to what was run specifically. Timestamps also show the Powershell_ise.exe binary was also accessed.

Recovered Certutil Logs reference the PowerShell modules *Connect-Mstsc.ps1, PSnmap.psd1* (both of which are the names of pre-made modules widely available online) and a more enigmatic 2sys.ps*1 ("Command Line: CertUtil -decode file.b64 2sys.ps1")* which shares a name to script referenced in a Carbon Black report on another ransomware group from 2018 (Dharma).

This usage of CertUtil is evidence of Remote File Copies (T1105).

MACB timestamps also suggest WMIC was used, although given the utility of this software its purpose cannot be confirmed. It may have been caused by some of the intruders tooling rather than directly invoked.

In addition to other precompiled binaries and tools *(7za, via certutil: CertUtil -decode za 7za.exe),*

the Certutil.log file once again offers some evidence of staging tooling; with the following commands executed (via script):

```
402.481.948: Begin: 17/06/2020 3:42 a.m. 43.278s

402.485.0: certutil.exe: $DOMAIN\$admin_account

301.3506.0: certcli.dll: 6.0:6002.18005 retail

301.3506.0: certutil.exe: 6.0:6002.18831 retail

301.3406.465: Command Line: CertUtil -decode arch.b64 arch.zip

301.3425.511: Command Succeeded

402.360.949: End: 17/06/2020 3:42 a.m. 43.292s
```

< trimmed for brevity>

```
402.481.948: Begin: 17/06/2020 3:42 a.m. 45.649s

402.485.0: certutil.exe: $DOMAIN\$admin_account

301.3506.0: certcli.dll: 6.0:6002.18005 retail

301.3506.0: certutil.exe: 6.0:6002.18831 retail

301.3406.465: Command Line: CertUtil -decode arch47.b64 arch.z47

301.3425.511: Command Succeeded

402.360.949: End: 17/06/2020 3:42 a.m. 45.658s
```

With a total of 48 numbered arch<$number>.b64 files decoded in sequence.

Based on the prior collection of 7za.exe (a command-line version of 7zip) and the fact the first of these files ended was decoded to have a .zip extension, it's a reasonably safe assertion that this was a multipart archive file.

Additional file structures beneath the C:\SQLDB\ folder were visible via shellbags, in particular folders called `kamikadze new` (as referenced here) `_local` and `77`.

The intruder installed a C++ Runtime library (Microsoft Visual C++ 2015 x86 Runtime 14.0.23026) to the system. It appears as though this or some of other parts of the software deployed prompted the system to collect a large number of Windows updates (such as KB2999226). These artefacts are inconclusive but were not exhaustively searched, as are the subsequent .msi's the system installed (their names variously available as deleted registry items) not long before executing the ransomware.
