Ryuk Ransomware: The Deviance is in the Variance

By Matthew FulmerManager, Cyber Intelligence Engineering

Published: 2020-11-24 · Archived: 2026-04-06 00:49:43 UTC

We have heard this story before, so I will skip the backstory about how ransomware is a giant nightmare that shows absolutely no sign of slowing down, at all. There is plenty of coverage out there pertaining to Ryuk, but lesser known is that there are different variants of Ryuk and in fact different generations of Ryuk which could potentially be hitting your environment.

Generation 1 of Ryuk was, to be polite, less than intelligent. A derivation of Hermes, which at its time had its own level of success, Ryuk was a new variant with an added new mechanism to drop the ransom payload. For this to succeed, the hacker groups were highly reliant on using tools they had access to, such as Trickbot, which is still considered a very notorious and highly successful trojan.

A gen1 encryption of Ryuk was rarely found without also finding Trickbot in the environment, which meant that if you could shut down Trickbot you would have a very successful chance of not getting hit with Ryuk. Why is this the case? Trickbot is how the group gained an initial foothold to launch their reconnaissance of the network. If the trojan is blocked at the gates, then there is no ground to attack.

A More Manual Version

What makes gen1 of Ryuk different from other ransomware variants is the human element. Once Trickbot is on the machine, someone needs to do the reconnaissance work to gain further access into the machine which would cause the most “impact” to the environment and thus force the hand to “guarantee” a payment.

Common targets would be critical machines such as fileservers, domain controllers, or hosting machines. Why was gen1 so successful, even with the increased ability to share information pertaining to malware/ransomware and countless sites where you can see live submissions from ample sources? Plain and simple, the ripest targets tend to be the ones which administrators “exclude” from strict AV policies to minimize the impact on end-users (read: performance delays, application crashes, etc.) or in some instances have the product outright disabled.

Next-Gen Ryuk

Now we have the next generation of Ryuk, where its creators have gone back to school, gained advanced knowledge on how to penetrate an environment and iterated. A pattern that admins everywhere do not want to hear about. While some things have changed (I will go over those shortly), other things have stayed the same. Basic entry to the environment still relies primarily on targeted spear-phishing and a “document” which most likely has a VBA macro included, and will execute the dropper/loader the second you enable macros (or, open the document in the case you have content enabled by default). There is also still a possibility that Trickbot can be part of the equation (after all, if it’s not broken, don’t fix it)

Common fare, many campaigns use this method. MAZE (now defunct), Ragnar, Emotet Malware, and more use this methodology as their first phase of gaining a foothold. Part of their advanced education has taught them about some new tools which can be leveraged, such as CobaltStrike Beacon, something commonly used during red-team penetration testing exercises because of the flexibility it gives to the one controlling the beacon (C2). Wizard Spider has started to rely substantially less on customized tools and instituted a new phase of living off the land, making their ability to laterally move once inside an environment exponentially easier.

Once they have a foothold and the C2 server in place, it's the same fare as other ransom attacks; leveraging command and Powershell for lateral movements and credential dumping. Unfortunately, there are not many products which are capable of monitoring Powershell scripting, substantially less can analyze Powershell scripts on a contextual level to allow legit scripts through and malicious scripts to be shut down.

This entails two options; either to outright block Powershell or to allow Powershell. Neither are great options if you want the utmost of security and balance for your users (specifically admins) to properly manage the environment or complete tasks. This harkens back to admins doing anything they can to not impact their user base, even if that means lowering the security posture of the environment.

An example of what can happen when you take that lowered security stance anywhere in the enterprise are the dozens of hospitals that have been hit with generation 1 of Ryuk at the height of the Coronavirus pandemic. A terrible situation which no healthcare provider should ever be caught in the middle of.

This is where prevention becomes essential. It's not an option anymore and no one can afford to deprioritize prevention as secondary to detecting and trying to remediate. In the case of ransomware, organizations can't afford any risk at all, in September of 2020 a patient died due to a ransomware attack at a hospital which prevented them from accepting new in-patients.

A colleague of mine made a video about Ryuk which shows prevention in action. In the video, the static engine brain version 109 is used to prevent Ryuk. This is important to note as the 109 brain was trained and released in November 2018 – two years prior to infection!

https://youtu.be/TOpIJa5Pf90

The above video helps to outline why a prevention-focused solution is absolutely necessary. The endpoint security solution provides intelligent Powershell script analysis that supports admins with the flexibility to do their work but simultaneously applies a rigorous prevention method against malicious actions. Deep Instinct really is the complete package.
