Final NSM Project

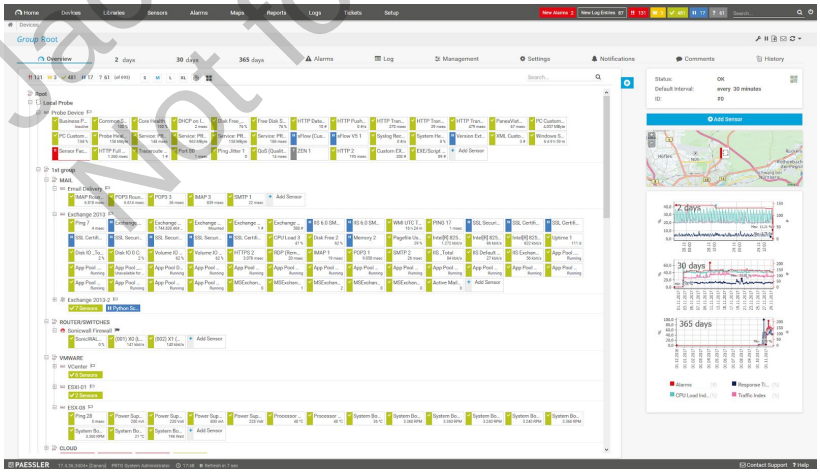
Jackson Nestler

University of Advancing Technology

Final NSM Project

For the final project, I chose to compare a freemium tool (not open source) against SecurityOnion. PRTG by Paessler is a network monitoring tool that can double as a security monitoring appliance, depending on your setup. PRTG requires a one-time license cost of anywhere from $1,600 to $60,000 depending on what you want to be able to set up. The ecosystem consists of sensors and servers, which are easily deployed through a Windows executable.

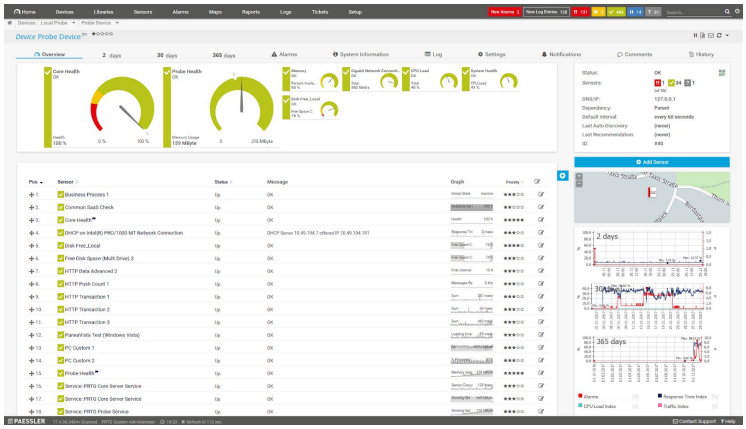
As advertised, PRTG promises to do network autodiscovery (through basic ping sweeps, so not effective if your organization has disabled ping responses), mapping of the network as it finds hosts, and alerting on "unusual" metrics or warnings. They allow for Android and iOS push notifications, SMS messages, email notifications, and even have an API that you can hook into and alert from that to an RSS feed, Slack channel, what-have-you. Their claim-to-fame is that they automate portions of a junior system administrator's job; take this for what you will.



The software works in a sensor/server setup as discussed previously, but it's not what you'd expect from a typical sensor/server array. There's a single server ("master" server) for most plans, but with some of the higher tiers ($45,000 and $60,000) you can have more than one server. Then, sensors suck up data from the network (through ping sweeps, passive monitoring, etc) and throw the data into a single dashboard on the server. SecurityOnion has a very similar setup, but it doesn't care how many "master" servers you have. With PRTG, you have to find a way to transfer the data between VLANs, physical sites, and even between cloud and on-prem instances. If you try to operate in a single-server setup, you're going to have a lot of security problems arise from working around the above problems.

The sensors perform actions on the network and feed that data into a dashboard. You can setup a "sensor" to ping a specific host at a specific time on certain days. You can setup a sensor to check if an SSL certificate is invalid. This is far more active than SecurityOnion.

How does it stack up against SecurityOnion? Not very well in the security perspective, but about as well in the network monitoring realm. PRTG passively looks at things like DNS traffic, plaintext HTTP requests, etc. It has the basic functionality of SecurityOnion, but not all the built in tools like Cyberchef, Bro, Elsa, or even some of the big components of the ELK stack. The graphs that PRTG forms by default are beautiful and will present well to an executive; SecurityOnion's ELK stack can have graphs that are just as beautiful, if not more, but it requires knowledge and setup; PRTG's comes out-of-the-box.



PRTG saw some malicious traffic but didn't consider it something worth alerting on (ie sending an email, text, etc). It did see it as a warning, and it looks like you could use their API to send emails, texts, etc on warnings too. The functionality is there, but needs to be implemented in a roundabout way. SecurityOnion took the cake for malicious[1] traffic though: it saw a bunch of DNS queries that were concerning. PRTG was setup on my home network (alongside SecurityOnion) and SecurityOnion would alert frequently on DNS queries, despite them being blackholed by my PiHole[2] instance. PRTG didn't care about these malicious DNS queries.

SecurityOnion misses the network discovery part, which could be scripted and fed into the ELK stack that SecurityOnion provides. There's also a lack of built-in alerting by SMS, email, etc. There is an API of sorts, but it's the ELK API.

Overall, I believe that PRTG is geared more towards systems administrators than it is for network security analysts. Both SecurityOnion and PRTG have their place and should be considered for deployment in your organization: I will continue to use PRTG and recommend its

[1] Malicious meaning that it was unwanted; this includes advertisements, trackers, and telemetry. This could be why PRTG didn't alert on the queries, whereas SecurityOnion did.

[2] A network-wide DNS blackhole; free open source software that prevents DNS queries from going out. If they're on a blacklist, they're redirected to 0.0.0.0 or 127.0.0.1 and go nowhere outside your network.

free version to anyone that's interested. However, during their 30 day premium trial I wasn't very impressed with the "premium" features (aka having unlimited sensors and a faster update path). Maybe this will change in the future, and I'm sure if I read the documentation a bit more I could find further uses for the premium features.