Internal Sinkhole

Studiengang: MAS | Vertiefung: MAS Information Technology

Aktuell wird nicht proxyfähiger Netzwerkverkehr direkt auf der Firewall blockiert und protokolliert. Die Informationen aus diesem Protokoll sind jedoch für die weitere Analyse zu wenig aussagekräftig. Um Malware oder fehlerhaft implementierte legitime Software besser identifizieren zu können, soll dieser Netzwerkverkehr gezielt in ein internes Sinkhole umgeleitet werden. Dort soll der Netzwerkverkehr aufgezeichnet und analysiert werden.

Ausgangslage

Heute wird nicht proxyfähiger Netzwerkverkehr auf der Default Gateway Firewall blockiert und protokolliert, mit dem Ziel, Einblick in den „unüblichen" Netzwerkverkehr zu erlangen. Dadurch werden Quell- und Ziel-IP-Adressen sowie die involvierten TCP/IP-Protokolle und Dienste zwar sichtbar, jedoch geben diese Informationen zu wenig Auskunft über die Absichten des Verbindungsaufbaus. Da die Verbindungen nicht zu Stande kommen, können die servicespezifischen Requests, respektive die Payloads, nicht aufgezeichnet werden. Unabhängig davon macht es Sinn, die Kommunikation die ausserhalb der Proxyserver stattfindet genauer zu untersuchen. Einerseits um fehlerhaft konfigurierte, legitime Software aufspüren und unnötigen Overhead im Netzwerk bereinigen zu können. Andererseits um mögliche Malware auf Geräten besser entdecken, respektive die weiteren Absichten der Sofware untersuchen zu können.

Zielsetzung

Ziel ist die Evaluation, Konzeption und Implementation eines internen Sinkholes zur Ergänzung der Default Gateway Konfiguration. Das Sinkhole soll Netzwerkverbindungen auf allen TCP-Ports entgegennehmen und beantworten können. Mittels Network Intrusion Detection System (NIDS) sollen die eingehenden Verbindungen analysiert und allenfalls Alarme generiert werden können. Zudem sollen alle Verbindungsdaten vollständig (Full Packet Capture) in PCAP-Dateien aufgezeichnet werden, um nötigenfalls manuell weitere Analyse betreiben zu können. Standardprotokolle wie DNS, HTTP und SMTP, welche auch von Malware häufig verwendet werden, sollen automatisch in separate Textdateien (Log) extrahiert werden. Alle neu gewonnenen Informationen sollen zudem zur weiteren Verarbeitung an die bereits bestehende Splunk Infrastruktur übermittelt werden. Dadurch soll der nicht proxyfähige Netzwerkverkehr effizienter analysiert werden können.

Vorgehen

Nachdem die Detailanforderungen in Zusammenarbeit mit den Bedürfnisträgern definiert wurden, konnten passende Softwarekomponenten und die Architektur evaluiert werden. Sobald der Lösungsweg klar war, wurde ein Proof-of-Concept in einer vom Unternehmensnetzwerk unabhängigen Laborumgebung durchgeführt. Dort wurde die Lösung ausgiebig und mit aktueller Malware (Emotet) getestet. Als Abschluss des Praxisteils wurde die Lösung an die produktiven Gegebenheiten adaptiert und im Betrieb implementiert.

Lösung

Das interne Sinkhole wurde auf dem im Unternehmen gebräuchlichen Red Hat Enterprise Linux 8 aufgebaut. Für die Verbindungsannahme wurde die Software INetSim von Thomas Hungenberg und Matthias Eckert eingesetzt. INetSim kann gerade bei Standardprotokollen wie FTP und HTTP mit Honeypot-Funktionen punkten. Als IDS-System wurde Suricata, entwickelt von der Open Information Security Foundation (OISF), ausgewählt. Nicht zuletzt, weil Suricata neben der Erfüllung sämtlicher Anforderungen auch bereits im Unternehmen vertreten war. Zusätzlich zum Extrahieren von Informationen aus den gewünschten Standardprotokollen in separate Log-Files, kann Suricata auch direkt Full Packet Capture (FPC) Aufzeichnungen in PCAP Dateien ausführen. Mittels Network Address Translation (NAT) auf der Default Gateway Firewall werden in Zukunft die suspekten IP-orientierten Verbindungen ins interne Sinkhole geleitet, um dort analysiert zu werden. DNS-orientierter Netzwerkverkehr wird mittels DNS Response Policy Zone (RPZ) in das Sinkhole gesendet. Der gewünschte Informationsgewinn konnte durch die Implementierung der Umgebung erreicht werden. Die fortlaufende Weiterentwicklung dieser trägt dazu bei, dass Ein- und Ausbruchsversuche im Netzwerk zeitnah erkannt werden können.

Dario Bürgi

Adrian Schuhmacher