Sambafreigaben in Univention/UCS vor Verschlüsselungstrojaner schützen

Du möchtest dich gerne für unsere Hilfe erkenntlich zeigen . Gerne. Wir bedanken uns bei dir für deine Spende!

Spenden

Zum frei verfügbaren Apt-Repository

GITLAB:

Getestet mit UCS4 und UCS5

Setzt man Enterprise Linuxserver wie UCS in seiner Umgebung ein sollte man nicht von der Annahme ausgehen das man vor Verschlüsselungtrojanern sicher ist. Denn hat man auf der Clientseite auch Windows am Laufen, und haben diese Windowsrechner auch Zugriff auf die Freigaben, ist das Prinzip des Verschlüsselns vom Client zum Server komplett das selbe. Und Ruckzuck kommt man schon wieder in Teufels Küche.

Da wir auf der Serverseite ein Linuxsystem verwenden ist es mit dem Tool Fail2ban sehr einfach Cryptorojaner mittels Auditlogs auszusperren. Wie das ganze unter UCS funktioniert, erkläre ich dir diesem Beitrag. Grundsätzlich ist diese Vorgehensweise unabhängig für jeden Sambaserver gültig.

Wie funktioniert das ganze?

Zum einen wird Samba so konfiguriert, dass es alle Aktivitäten mitlogt. Genauer gesagt das Schreiben und Umbenennen von Dateien. In den Logs findet sich dann ein Eintrag wie dieser:

```
Mar 11 10:44:33 fileserver smbd_audit:
IP=192.168.4.211|USER=joe|MACHINE=desktop-
joe|VOLUME=public|pwrite|ok|zaehlerstaende - Kopie.locky
```

Im zweiten Step weisen wir Fail2Ban mit vordefinierten RegEx an, dass er sich diese Logs anschauen und nach Logeinträgen wie diese untersuchen soll. Wird ein Eintrag gefunden, so ist der Client-Computer sofort mittels IPTABLES-Regel vom Server zu entfernen und dem Administrator eine Email zuzuschicken. Eine vordefinierte Bantime ist groß genug gewählt, dass der Administrator genug Zeit zum Reagieren hat.

Filter für „grep“:

```
overwrite_if
file_id_create
create_file
```

```
pwrite
```

Installation

Diese gestaltet sich recht einfach, und muss auf dem Fileserver passieren. Fail2ban ist nicht in den unterstützten Quellen enthalten, daher müssen zuerst die unmaintained freigeschalten werden.

```
ucr set repository/online/unmaintained=yes
apt update
apt install fail2ban -y
```

Das war's auch schon. Fahren wir nun mit der Konfiguration fort.

Konfiguration Samba

Wir öffnen die Datei `/etc/samba/local.conf` und fügen folgenden Inhalt in die **[GLOBAL]** Sektion ein:

```
[GLOBAL]
full_audit:failure = none
full_audit:success = pwrite write rename create_file
full_audit:prefix = IP=%I|USER=%u|MACHINE=%m|VOLUME=%S
full_audit:facility = local7
full_audit:priority = NOTICE
```

Unter UCS5 ändert sich das ganze ein wenig, hierbei wird `rename zu renameat`.
Und Samba neu starten:

```
/etc/init.d/samba restart
```

Und für jede Freigabe die überwacht werden soll: vfs objects = full_audit. Dies würde für einen Standard Samba auf Ubuntu genügen. UCS hat für „vfs objects“ default bereits ein Modul geladen, deshalb sieht unser Eintrag in UCS etwas anders aus: `vfs objects full_audit acl_xattr`
Eingetragen wird das ganze in der UDM (Univention Directory Manager) unter „Domäne → Freigaben → Freigabe auswählen“, „Erweiterte Einstellungen → Erweiterte Samba-Einstellungen → Schlüssel: **vfs objects**, Wert: **full_audit acl_xattr**.

Fail2Ban-Konfiguration

Wir brauchen dazu zwei Teile: Zum einen die Datei in der steht auf was genau Fail2Ban achten soll und in der anderen Datei welche Regelungen für diesen Audit gelten. Zunächst erstellen wir die Datei `/etc/fail2ban/filter.d/samba.conf` und füllen sie mit folgendem Inhalt:

```
[Definition]
```

```
failregex = smbd.*\:\ IP=<HOST>\|.*\.locky$
            smbd.*\:\ IP=<HOST>\|.*_Locky_recover_instructions\.txt$

ignoreregex =
```

Dies hier ist nur ein Beispiel. Eine Datei mit den Ende 2020 bekannten Definition habe ich hier angehängt:

samba.conf

Nun erstellen wir die Datei `/etc/fail2ban/jail.d/samba.conf` und füllen auch diese mit dem Inhalt:
Der Loggingpfad kann je nach Funktion abweichen.

```
[samba]
filter = samba
enabled = true
action = iptables-multiport[name=samba, port="135,139,445,137,138",
protocol=tcp]
         mail[name=samba, dest=admin@MYDOMAIN.DE]

# Einträge in die Syslog werden beobachtet
logpath = /var/log/remote-logging/data/smbd_audit.log
# Die erste umbenannte oder erstellte Datei mit der
# Endung .locky führt zum Ausschluss vom Fileserver
maxretry = 1
# Die letzten 10 Min des Logs werden berücksichtigt
findtime = 600
# Der Client ist für einen Tag ausgesperrt
bantime = 86400
```

Nun noch Fail2Ban neu starten und der Schutz ist aktiv. Man beachte, dass in jedem Fall eine Datei umbenannt werden wird bevor der Schutz greift. Der Eintrag in die Log geschieht leider nun mal erst nach der Aktion. Somit ist diese eine Datei leider verschlüsselt.

Das Kreuz mit der Groß- und Kleinschreibung

Jetzt haben wir uns so schön überlegt wie wir Dateien auf dem Fileserver schützen. Blöde ist nur, dass das System Unterschiede zwischen .locky und Locky und erst Recht .LOCKY macht. Die Entwickler der Cryptotrojaner sind ja auch nicht doof. Man könnte sich nun die Mühe machen jede mögliche Kombination in die `/etc/fail2ban/filter.d/samba.conf` einzutragen. Nimmt man dann noch die „1337“-Schrift dazu, dann erhöht sich das noch mal imens.

In diesem Fall sind reguläre Ausdrücke dein Freund. Wir geben einfach Fail2Ban die möglichen Buchstaben mit aus denen .locky gebildet werden könnte. Beispielsweise deckt `[lL][oO][cC][kK][yY]` alle Kombinationen aus Groß- und Kleinschreibung ab. Fail2Ban geht automatisch alle Möglichkeiten durch. In der `/etc/fail2ban/filter.d/samba.conf` müsste das dann so ausschauen:

```
smbd.*\:\ IP=<HOST>\|.*\.[lL][oO][cC][kK][yY]$
```

Testen

Nun kopiert man vom Client aus eine Datei auf den Fileserver, die mit einer von diesen Erweiterungen. Dies kann eine einfache Textdatei sein. Fail2Ban sperrt nun den Zugriff auf das Netzlaufwerk. Die eine Datei ist allerdings auf dem Zielort abgelegt worden. Nur eben keine weitere.

Ob die Sperrung funktioniert hat sieht man mit `iptables -nvL` oder auch mit `iptables -nvL –line-numbers`

```
Chain f2b-samba (1 references)
num  target     prot opt source               destination
1    REJECT     all  --  192.168.110.206      anywhere             reject-
with icmp-port-unreachable
2    RETURN     all  --  anywhere             anywhere
```

Möchte man die Sperre wieder aufheben, so macht man das mit folgender Zeile: `fail2ban-client set samba unbanip 192.168.110.206`
Oder direkt mit IPtables: `iptables -D f2b-samba 1` Wobei du bei dieser Methode beachten musst, dass bei einem Neustart von fail2ban der Block in den IPtabeles wieder aktiviert wird.

fail2ban-client ist grundsätzliches ein sehr nützliches Tool. z.B. `fail2ban-client status`