

Intro aux enjeux techniques du cyberspace

Chapitre 4 Étude de cas : Stuxnet



Pablo Rauzy <pr@up8.edu>
pablo.rauzy.name/teaching/ietc

Étude de cas : Stuxnet

- ▶ Stuxnet est un *ver* informatique pour le système Windows.
- ▶ Il a été découvert en 2010 par une société de sécurité informatique biélorusse.
- ▶ Il ciblait centrifugeuses iraniennes d'enrichissement d'uranium.
- ▶ Il aurait contaminé 45000 systèmes dont 30000 en Iran.
- ▶ Il s'agirait de la première *cyber arme* conçue pour attaquer un système industriel précis.

- ▶ Un *ver* est un logiciel malveillant capable de se propager à travers un réseau informatique.
- ▶ Au contraire d'un virus il n'a pas besoin d'un programme "hôte" (autre que le système).

- ▶ Stuxnet procède en 3 étapes :
 - Infecter des Windows jusqu'à en trouver qui contiennent les logiciels Siemens PCS 7.
 - Infecter le logiciel Step 7 (interface utilisateur du *SCADA* visé).
 - Infecter les machines Siemens S7 PLC contrôlées par ce logiciel.

- ▶ Acronyme de “Supervisory Control And Data Acquisition”.
- ▶ Système de gestion qui prend des décisions en temps-réel en fonction de mesures.

Étape 1 : Windows

- ▶ Stuxnet s'introduit sur les réseaux internes et saute les *air gap* par clef USB.
- ▶ Il s'installe sur les systèmes Windows grâce à un *Oday* (CPLINK).
- ▶ Il se diffuse à travers le réseau en utilisant différents *exploits* et Odays (partage d'imprimantes, LNK/PIF, ...).
- ▶ Il contient deux *rootkits* un au niveau utilisateur et un au niveau noyau.
- ▶ Il contient un *pilote* qui a été *signé* par des certificats volés à JMicron et Realtek (deux entreprises sur le même campus à Taiwan).
- ▶ Deux sites web lui servent de *C&C* (au Danemark et en Malaisie).
- ▶ Il est très gros pour un malware (un demi mégaoctet), et programmé en plusieurs *langages* (notamment C et C++).

Air gap

- ▶ On parle d'*air gap* quand un système ou un réseau est physiquement déconnecté de tout autre réseau.
- ▶ D'où la nécessité d'utiliser une attaque via clef USB pour y pénétrer.
- ▶ Stuxnet était donc prévu pour attaquer un réseau sensible.

- ▶ Un *0day* est une vulnérabilité qui n'a jamais été divulguée.
- ▶ Son utilisation dans un malware le révèle le plus souvent quand celui-ci est analysé.
- ▶ Stuxnet exploite quatre 0day différents, ce qui est complètement inhabituel.

- ▶ Un *exploit* est un composant d'un programme permettant d'exploiter une vulnérabilité.
- ▶ Il peut s'agir d'un exploit distant pour se connecter à un autre système, ou d'un exploit local pour accéder à des privilèges supérieurs par exemple.
- ▶ Stuxnet utilise plusieurs exploits réseaux et locaux.

Rootkit

- ▶ Un *rootkit* est un ensemble de technique visant à pérenniser et cacher la présence d'un malware dans un système.
- ▶ Ce sont des logiciels assez complexes.
- ▶ Ce n'est jamais utilisé avec de bonnes intentions.
- ▶ Stuxnet en utilise plusieurs à différents niveaux et sur différents types de système.

- ▶ Programme dont le rôle est de permettre l'interaction entre un logiciel et un périphérique.
- ▶ Quand un pilote doit s'installer dans un système critique, il est parfois requis que celui-ci soit signé.
- ▶ Stuxnet embarque un pilote pour intercepter et altérer les communications entre le logiciel de contrôle et supervision des centrifugeuses et ces dernières.

- ▶ Une *signature* est une garantie cryptographique de l'authenticité et de l'intégrité d'une donnée (qui peut-être un logiciel).
- ▶ Cela suppose l'utilisation de cryptographie asymétrique, c'est à dire une clef privée qui permet de faire la signature et une clef publique qui permet de la vérifier.
- ▶ Stuxnet embarque deux signatures valides par deux entreprises différentes pour son pilote, ce qui signifie que les clefs privées ont été volées.

- ▶ C&C est l'abréviation de "*command and control*".
- ▶ C'est comme ça qu'on désigne le serveur maître d'un parc de systèmes infestés par un rootkit.
- ▶ Il peut servir à contrôler un botnet, ou à collecter des informations, par exemple.
- ▶ Stuxnet utilisait un C&C pour être mis à jour et pour de l'espionnage industriel.

Langage

- ▶ On parle ici de *langage de programmation*.
- ▶ La plupart des malwares n'en utilisent qu'un seul.
- ▶ Stuxnet utilise plusieurs langages, ce qui peut laisser penser qu'il a été développé par plusieurs équipes.

- ▶ Une fois sur un Windows, Stuxnet infecte les fichiers des logiciels WinCC et PCS 7.
- ▶ Il utilise entre autre un 0day pour accéder à la *base de données* du logiciel.
- ▶ Il peut alors modifier la configuration du logiciel.
- ▶ Il subvertie la *bibliothèque* “*s7otbxdx.dll*” qui sert à la communication entre WinCC et les Siemens S7 PLC.
- ▶ Il peut alors intercepter les communications et les altérer de manière masquée.
- ▶ Ça lui permet de s’installer sur les PLC de manière invisible, et de masquer sa présence.

- ▶ Une *base de données* est littéralement ce que ça décrit.
- ▶ Stuxnet utilise un 0day qui lui permet de lire et d'écrire dans la base de données de Step 7 (un mot de passe écrit en dur dans le code, ce qui laisse penser que le logiciel coopère).

- ▶ Une *bibliothèque* (ou *librairie* par anglicisme) est un ensemble de composants qu'on utilise dans un ou plusieurs logiciels et qui peut être développée séparément de ceux-ci.
- ▶ Stuxnet modifie la bibliothèque qui s'occupe de la communication avec les centrifugeuse, pour faire une attaque de l'intercepteur.

Étape 3 : Siemens S7 PLC

- ▶ Étape moins connue car le code n'a pas été publié.
- ▶ Visiblement le malware n'agit pas sauf si il reconnaît une configuration précise.
- ▶ Quand il agit, il modifie périodiquement la vitesse de rotation des moteurs du système.
- ▶ Il installe également un rootkit pour se rendre invisible.