System Administration: RAT

D. Leeuw

12 december 2023
v.0.1.0

Over dit Document

0.1 Leerdoelen

Na het bestuderen van dit document heeft de lezer:

- een idee hoe een systeem op afstand beheerd kan worden
- wat ervoor nodig is om een systeem op afstand te beheren
- kennis genomen van Ansible als voorbeeld van beheer op afstand

0.2 Voorkennis

Om dit document goed te kunnen begrijpen is de volgende kennis vereist:

- public key infrastructure (PKI) en cryptografie
- shell scripting en/of scripting in powershell

Inhoudsopgave

Hoofdstuk 1

RAT - Introductie

RAT is een afkorting met een aantal mogelijke betekenissen, namelijk Remote Access Tool, Remote Administration Tool en Remote Access Trojan. Remote Access is een heel breed onderwerp, waar elke vorm van toegang tot systemen over een netwerk onder valt. In dit document zullen we ons beperken tot software waarmee we over een netwerk toegang tot een systeem kunnen krijgen om de configuratie van een machine aan te passen, Remote Administration dus. Remote Access Trojans worden behandeld in het "Security: Malware" document onder trojans.

Remote Administration is iets dat veel door systeembeheerders wordt gebruikt om servers in de serverruimte te configureren zonder dat er elke keer naar de serverruimte gelopen hoeft te worden. Er zijn ook software pakketten die het mogelijk maken om een compleet netwerk te voorzien van de juiste configuratie, dus ook voor het aansturen van routers en switches.

Remote Administration kan ook gebruikt worden door de cybercriminelen om machines remote te "beheren". Deze malafide RAT software verbergt zich vaak op het systeem en is moeilijk terug te vinden. We kunnen deze software aantreffen op computers, maar ook op printers, smartphones of IoT-devices.

Hoofdstuk 2

Web Interfaces

De makkelijkste en waarschijnlijk de bekendste manier is het configureren van bijvoorbeeld een access point of router via een web-interface. Je logt in met een gebruikersnaam en wachtwoord op een website en krijgt via die website toegang tot de configuratie van het apparaat.

Vanuit beveiligingsoogpunt is het van belang dat de website gebruik maakt van HTTPs. HTTP is ongeencrypt verkeer, dat houdt in dat alle data in leesbare tekst over de lijn gaat, dus ook je gebruikersnaam en wachtwoord. Iemand die meeluistert op de lijn kan deze gegevens dus zien en kan met jouw credentials het apparaat gebruiken. Dus moet je altijd HTTPs gebruiken, die encrypt de data die over de lijn gaat waardoor jouw credentials niet voor anderen leesbaar zijn.

Mocht je een nieuwe apparaat hebben dat niet met HTTPs werkt, log dan in met de meegeleverde credentials, zet de webserver op HTTPs (eventueel met een selfsigned certificate) en wijzig daarna alle wachtwoorden. Eventuele gelekte wachtwoorden zijn dan niet meer te gebruiken en de nieuwe wachtwoorden kunnen niet meer gelekt worden.

Hoofdstuk 3

Remote Desktop

Microsoft Windows gebruikt een grafische interface om de gebruiker toegang te geven tot de functies van een computer. Als er een tool zou zijn die het scherm van de server kopieert naar het netwerk (naar de beheermachine) en de muis bewegingen en toetsaanslagen van de beheermachine naar de server stuurt, dan kunnen we op afstand een grafische interface beheren. Als we zo ver zijn kunnen we natuurlijk ook bij de configuratie van de machine komen.

Er zijn heel veel verschillende software pakketten die deze functionaliteit bieden. Wij zullen er in dit hoofdstuk 2 behandelen: RDP en VNC. Niet dat we een voorkeurd hebben voor de ene of andere, het gaat om de makkelijke toegankelijkheid van deze software.

3.1 RDP

De afkorting RDP staat voor Remote Desktop Protocol het is het protocol dat tussen de client en de server wordt gebruikt. De server kan ook een ander desktop OS zijn, voor het gemak praten we over de server als we het hebben over de machine die de desktop aanbiedt, de client is dan de machine de een verbinding maakt met de server. Op de client moet software draaien die het Remote Desktop Protocol praat. We noemen de software op de client de Remote Desktop Client of sinds Windows 10 Remote Desktop Connection, de afkorting is voor beide hetzelfde: RDC. Het geheel van server software, client software en het netwerk protocol heet de Remote Desktop Service, afgekort: RDS.

Zoals de naam al aangeeft is dit meer een desktop op afstand, dan een enkele applicatie. Met RDS neem je de complete desktop over en daarin kan je applicaties opstarten.

RDS is ooit begonnen als Windows Terminal Server en is tegenwoordig bekend onder de naam Remote Desktop Connection.

Het RDP (Remote Desktop Protocol) luistert standaard op port 3389. Het protocol is ontwikkeld door Microsoft op basis van het ITU T.128 protocol. De versie van Microsoft is gepatenteerd door Microsoft. Het protocol geeft de mogelijkheid tot het overnemen van systemen en het meekijken bij en het ondersteunen van gebruikers.

RDP geeft de gebruiker de mogelijkheid om ook geluid naar de remote desktop te sturen. Lokale bestanden, printers en interfaces kunnen gebruikt worden op de remote desktop. Ook knippen en plakken tussen de lokale en de remote omgeving is mogelijk.

3.2 VNC

VNC (Virtual Network Computing) is een client/server architectuur. De server draait op de machine waarvan het scherm overgenomen moet worden. De client is de ontvanger van de scherm informatie.

Het VNC-protocol gebruikt blokjes om de scherm informatie over het netwerk te sturen. Als eenmaal het volledige beeld is opgebouwd zullen alleen de blokjes worden overgestuurd waar zich een wijziging heeft voorgedaan. Om een veilige verbinding op te bouwen met VNC is het nodig VNC te tunnelen door een VPN. Als het alleen op het lokale netwerk wordt gebruikt vallen de veiligheidsrisico's over het algemeen mee.

Since versie 2 (2004) wordt het VNC-protocol ook gebruikt door de Apple Remote Desktop (ARD) applicatie. Hiermee is het mogelijk om met de ARD-client Windows of Linux systemen te beheren die een VNS-server hebben of met elke willekeurige VNC-client een Mac OS X systeem te beheren die een ARD-server heeft.

Hoofdstuk 4

Remote Command Line

Het over de lijn sturen van scherm informatie is natuurlijk een behoorlijke belasting van het netwerk en niet geschikt voor bijvoorbeeld switches of routers die geen grafische interface hebben. Een al heel oude oplossing hiervoor is het gebruik van de CLI (Command Line Interface) om systemen op afstand te beheren. De eerste oplossing hiervoor was het `telnet` commando. `telnet` werd gebruikt om routers en unix-servers te beheren. Het programma is echter nooit geschreven met security in gedachten. Alle commando's en de resultaten gaan in tekst over het netwerk en is dus theoretisch af te luisteren. Ook tijdens het inloggen gaat alles via tekst over de lijn. Gebruikersnamen en wachtwoorden zijn dus simpel te sniffen.

Het is dan ook de bedoeling dat `telnet` niet meer gebruikt wordt. De wel veilige opvolger is `ssh` wat een afkorting is voor secure shell en dus wel veilig is. `ssh` maakt gebruik van encryptie waardoor alles geencrypt over het netwerk gaat.

Het `ssh` protocol is een client-server-protocol. Er moet dus een stuk software draaien op de host waarmee je verbinding wil maken en op het beheerstation moet de ssh-client geïnstalleerd staan. Standaard draait de ssh-server op port 22. De client bouwt dan ook standaard een verbinding op met deze port. Draait de server op een andere port, dan moet dat aan de client verteld worden.

4.1 OpenSSH

OpenSSH (`https://www.openssh.com/` is een opensource software pakket dat zowel een SSH-server als een SSH-client bevat. OpenSSH is ooit (door)ontwikkeld door OpenBSD ontwikkelaars om een volledig opensource SSH-variant te ontwikkelen zonder patenten.

De meeste op Unix-gebaseerde systemen zoals Linux en Mac OS X zijn standaard uitgerust met een ssh-server of kunnen simpel van een ssh-server voorzien worden.

Om SSH-server op een Mac OS X machine aan te zetten verwijzen we graag naar de website van Apple: https://support.apple.com/lt-lt/guide/mac-help/mchlp1066/mac

Microsoft biedt OpenSSH aan via zijn eigen repository. Op de site van Microsoft staat beschreven hoe je je Windows Machine kan voorzien van OpenSSH: https://learn.microsoft.com/en-us/windows-server/administration/openssh/openssh_install_firstuse

SSH gebruikt standaard een username en password om in te kunnen loggen, dat is natuurlijk voor remote beheer niet handig. Een andere optie die de OpenSSH server kent is inloggen via een public en private key. De private key blijft natuurlijk geheim en is eigendom van de beheerder die bij de server moet mogen. De public key(s) wordt geplaatst op de server die men wil beheren. Als iemand wil inloggen op de server en de SSH-client aanroept dan zal deze wat informatie versleutelen met de private key. De server ontvangt dit bericht en zal het decrypten met de public key. Lukt dit dan weet de server zeker dat de inlogger de gene is met de private key. Op deze manier hoef je niet elke keer een username en password in te typen maar kan je op de commandline aan de SSH client meegeven welke private key er gebruikt moet worden.

Hoofdstuk 5

Orchestration

Orchestration maakt het mogelijk om verschillende machines te beheren waarbij bijvoorbeeld afhankelijkheden worden meegenomen. Een server kan zo vanaf scratch opgebouwd worden door de orchestration software. Eerst wordt het OS geïnstalleerd en als dat klaar is kan er bijvoorbeeld de webserver software geïnstalleerd worden. Is de webserver goed geïnstalleerd dan kunnen de verschillende configuraties voor de verschillende websites op de server gedownload worden.

Op een beheerstation (meestal de desktop van de systeembeheerder) wordt aan de orchestration software verteld wat er allemaal nodig is 1 of meerdere servers in te richten en up-to-spec te houden. Up-to-spec is dat de orchestration software ervoor zorgt dat de configuratie blijft zoals deze beschreven is. Als een beheerder dus op de machine zelf de configuratie wijzigt dan zal de orchestration software die wijzigingen weer terug zetten naar zoals dat door de beheerder eerder in de orchestration software bescherven is.

Dit alles kan gedaan worden door remote commando's te laten uitvoeren door de orchestration software. Dit kan bijvoorbeeld via SSH, maar de orchestration software kan ook een eigen client hebben die eerst op de machine geïnstalleerd moet worden waarbij er tussen de client en de server (LET OP: het beheerstation is nu de server geworden, ondanks dat het waarschijnlijk een desktop machine is) een eigen taal gebruikt wordt om te zorgen dat de client gaat doen wat de server wil.

De meest bekende tool is waarschijnlijk Ansible. Andere opties zijn Puppet, Chef, Salt Stack en nog vele anderen. Wij zullen alleen Ansible behandelen.

5.1 Ansible

Ansible maakt gebruik van SSH om verbinding te leggen met de verschillende te beheren systemen en gebruikt scripts (shell, powershell) om commando's uit te voeren. Hiermee heeft Ansible gelijk een voorsprong op andere tools omdat er alleen de kennis vereist is van SSH en een scriptingtaal die toch al veel door beheerders gebruikt wordt. Daardoor is de leercurve voor Ansible laag en dus de adoptiegraad hoog.

5.1.1 YAML

Index