IT-Intrånget vid D-Sektionen den 1/12 2021

Detta är en write-up av intrånget, och en händelse som skedde i anslutning till det. Jag har försökt beskriva relevanta tekniska begrepp, och förklara dem. Här följer en beskrivning av händelserna och vad vi kommer göra i framtiden:

Vid 13:45 mottog jag, i egenskap av root, ett mail från Martin Sunnerdahl vid LDC (Lunds DataCentral) om att en av vår servrar, hyacinth, visade tecken på att den var hackad. Den hade då kommunicerat med en IRC (Internet Relay Chat)-server sedan 12:24, och gjort konstanta nickname-changes. Detta hade lett till en flaggning i deras system. Då jag mottog mailet bekräftade jag att hyacinth hade en anslutning till den servern på port 443, och tog därefter beslutet att stänga av servrarna för att stoppa anslutningen innan jag visste mer om den. Eftersom jag fruktade en ransomware-attack stängde jag av filservern genom att dra ut strömmen då vissa ransomware-program börjar kryptera boot-filer först då ett shutdown-kommando utfärdas, för att undgå upptäckt tills det är för sent. Sedan stängde jag av de övriga maskinerna som vanligt, då de inte innehåller någon data vi riskerar förlora.

Webbsidan och vår funktion för att ta emot mail, samt inloggningar till hemsidan, förblev påslagen då den servern är ansluten till ett annat nätverk och definitivt inte var inblandad. Hyacinth fungerar som en brandvägg och router, varpå jag inte kunde avgöra vilken av maskinerna som satt bakom den var skyldig. Därför var jag tvungen att stänga ner alla maskiner på det nätverket, inklusive alla skärmar i iDét. Innan denna incident inträffade hade vi inte satt upp en tillräckligt utförlig loggning av paket, varpå vi inte kunde använda en sådan logg för att identifiera den infekterade maskinen.

Med hjälp av loggar från LDC fick vi reda på att den skyldiga maskinen hade kört två kommandon över nätverket innan den öppnade sin anslutning mot IRC-servern; curl och wget. Curl används för att hämta innehåll från webbservrar i textform, och wget för att ladda ner motsvarande innehåll. Den hämtade och exekverade ett perl-script som numera finns på https://pastebin.com/j1W7Jv0n för den som vill läsa igenom det. Scriptet verkar vara en version av en cirka 10 år gammal shell-bot som kretsat på internet. När det laddades ned verkar det ha lagts i /tmp, vilken är en mapp som rensas vid omstart i de flesta operativsystem. Det innebär att scriptet med största sannolikhet skulle vara borta när jag startade maskinerna igen.

Loggarna gav oss även att maskinen i fråga hade version 1.17.1 på wget och version 7.59.0-DEV på curl. Därför fick vi vår första metod för att kontrollera vilken maskin det var som var infekterad. Vi startade hyacinth och kontrollerade att den inte hade de aktuella versionerna av curl och wget med kommandot:

"$(which -a wget | xargs -I { } sh -c "{} –version" | grep -c 1.17.1 2> /dev/null)" och

"$(which -a curl | xargs -I {} sh -c "{} –version" | grep -c 7.59.0 2> /dev/null)".

Hyacinth var inte den skyldiga maskinen, varpå min företrädare Nils Ceberg satte upp en utförlig loggning av alla nya TCP-anslutningar och alla UDP-paket som går via hyacinth för att kunna spåra varifrån anslutningar kommer. Jag gjorde om ovanstående till ett script som via ansible skulle kontrollera alla våra maskiner, och se vilken som var skyldig. Sedan började vi starta alla maskiner, och körde scriptet.

Resultatet var att en av våra docker-containers var den enda möjliga maskinen baserat på

versionerna av curl och wget. Den körde en version av Gitlab som inte var uppdaterad på länge, då vi inte använder den. Vårt avlagda system Mattermost brukade använda Gitlab-containern som ett mellansteg mellan den och Ldap för autensiering. I maj presenterade Gitlab en CVE (Common Vulnerabilities and Exposures) - CVE2021-22205, vilken innebar att utomstående parter kunde få Gitlab-servrar att utföra kommandon som skickades till dem – utan att logga in. Vi har nu i efterhand upptäckt att vi inte är med i Gitlabs maillista för CVE:s, och inte heller någon lista som skickar vidare dem som vi trodde vi var. Därför upptäckte vi inte denna CVE, och förblev sårbara mot den då vissa av våra docker-containers dessutom inte autouppdateras.

En så kallad crawler-bot hittade denna sårbarhet hos oss, och lyckades få oss att ladda ned och köra perl-scriptet. På grund av att de kom in via en CVE, och inte med användaruppgifter, var alla sektionsmedlemmars data säker under hela intrånget. Dessutom är det bara inloggnings-uppgifter och personuppgifter till personer med user.dsek.se-mailadresser, som exempelvis styrelsen, som lagras på servrarna i det påverkade nätverket. Lösenord och användaruppgifter från hemsidan ligger på en annan server i ett annat nätverk. Alla lösenord lagras hashade och saltade. Men inga användaruppgifter eller lösenord har varit i fara. Gitlab-containern har dessutom inte tillgång till några inloggningar från vår nya LDAP-server, utan hade bara den gamla som har varit avstängd i flera månader.

Efter att ha dissekerat perl-scriptet som kördes på vår server har jag kunnat utläsa att det är till för att skicka paket-floods till specifika enheter. Det har ett flertal kommandon, men de som det verkar vara mest fokus på är till för att skicka TCP-floods och UDP-floods till IP-adresser som skickas till den via IRC-chattkanalen. IRC använder den för att ta emot kommandon, och visa att den är aktiv och har infekterat en dator. Om man ansluter till den kan man se miljontals keepalive-kommandon från olika enheter, vilket tyder på att detta botnet är relativt stort. Baserat på IP-adresser för servrar samt kommentarer i scriptet kan man utläsa att det förmodligen har sin bas i Brasilien. Jag har skickat mail till alla berörda internetleverantörers abuse-mailboxar, i hopp om att störa deras operation.

I framtiden:

För att undvika att liknande händelser uppstår har vi satt upp bättre loggning av nätverkstrafik, och funderar på att installera ett program för att övervaka trafiken och kunna få varningar baserat på potentiellt skadliga trafikmönster. Vi ska även se över vilka CVE-listor vi är med i, för att garantera att vi är med i alla som vi potentiellt kan påverkas av. Dessutom kommer våra dockercontainers få autouppdateringar, via en container som heter Watchtower, i större utsträckning.

Ryktesspridning:

Runt samma tid som intrånget inträffade så uppstod det en situation då jag såg en person som betedde sig misstänksamt i iDét, och stod och tittade på dörren till vårt fläktrum och verkade försöka dölja något. Nils skulle sedan jogga till en kortläsare för att aktivera sitt kort, varpå han ser samma person jogga mot utgången söder om iDét. Nils var inte medveten om att jag sett någon uppföra sig på ett konstigt vis en stund tidigare. Denna person vänder sig om då han hör Nils jogga bakom sig, och börjar då springa i full fart. Han ställer sig sedan en bit utanför dörren och tittar tillbaka på Nils, varpå han försvinner in bakom en bil. När Nils sedan berättade

historien för mig då han kommit tillbaka till iDét lägger vi ihop våra historier, och bestämmer oss för att ringa universitetets väktartjänst för att berätta om händelsen då vi bedömde att det var tillräckligt misstänksamt beteende för att behöva ringa. En väktare kommer då och lyssnar på vår historia och undersöker var i iDét personen befann sig. När detta skedde kommer även PH och talar med oss, delar av styrelsen och väktaren.

De kommer fram till att det potentiellt varit en person som undersökte om det fanns något värt att stjäla i iDét som är lätt att ta med sig. Detta då det varit en del stölder på campus. Inget är dock säkert. Under hela processen talade ingen med personen vi såg, och ingen hade en kniv på sig. Varken vi eller personen i fråga uttalade några hot – verbala eller fysiska.

Lund, dag som ovan

Oskar Stenberg
root