AD CS Abuse

Active Directory Certificate Services (AD CS)

Active Directory Certificate Services (AD CS) ဆိုတာက အဖွဲ့အစည်းကြီးတွေ၊ ကုမ္ပဏီတွေမှာ **digital certificate** တွေကို စနစ်တကျ စီမံခန့်ခွဲနိုင်အောင်လုပ်ပေးတဲ့ Microsoft ရဲ့ service တစ်ခုပါ။ Digital certificate ဆိုတာက လူတစ်ယောက်၊ ကွန်ပျူတာတစ်လုံး၊ သို့မဟုတ် server တစ်ခုဟာ “ဒီသူ အမှန်ပဲ” လို့ သက်သေပြပေးနိုင်တဲ့ အထောက်အထားတစ်ခုလိုပါပဲ။ အွန်လိုင်းမှာ အချက်အလက်တွေကို လုံခြုံစွာပို့ဆောင်ဖို့၊ login ဝင်တဲ့အချိန် user အမှန်လား မမှန်လား စစ်ဖို့၊ ဒေတာတွေကို ခိုးမယူနိုင်အောင် encryption လုပ်ဖို့ စတဲ့အရာတွေမှာ certificate တွေကို အသုံးပြုပါတယ်။

AD CS ရဲ့ အားသာချက်အကြီးဆုံးက Active Directory နဲ့ အလွန်နီးစပ်စွာ ချိတ်ဆက်ထားတာပါ။ Active Directory ဆိုတာက ကုမ္ပဏီတစ်ခုအတွင်းက user တွေ၊ computer တွေ၊ group တွေကို အလယ်ဗဟိုက နေ စီမံခန့်ခွဲတဲ့ စနစ်ဖြစ်ပါတယ်။ AD CS ကို Active Directory နဲ့ ချိတ်ထားလိုက်တဲ့အခါ certificate ထုတ်ပေးတာ၊ သက်တမ်းကုန်သွားတဲ့ certificate ကို ပြန်လည်သုံးစွဲနိုင်အောင် ပြင်ဆင်တာ၊ ဘယ် user က ဘယ် certificate ကို ရသင့်တယ်ဆိုတာ ဆုံးဖြတ်တာတွေကို အလိုအလျောက် လွယ်ကူစွာလုပ်နိုင်ပါတယ်။ ဒီလိုလုပ်နိုင်တဲ့အတွက် အချိန်ကုန်သက်သာပြီး စီမံခန့်ခွဲရတဲ့ အလုပ်တွေကိုလည်း လျော့ချပေးနိုင်ပါတယ်။

ဒါပေမယ့် ဒီလို လွယ်ကူအောင် ချိတ်ဆက်ထားတာကပဲ လုံခြုံရေးအရ အန္တရာယ်ဖြစ်လာနိုင်ပါတယ်။ အကြောင်းကတော့ AD CS ကို configuration မမှန်ကန်ဘဲ တပ်ဆင်ထားမယ်၊ permission တွေကို မသေချာစွာ သတ်မှတ်ထားမယ်ဆိုရင် attacker တွေအတွက် အခွင့်အရေးတွေ ပေါ်လာနိုင်လို့ပါ။ Misconfiguration ဆိုတာက စနစ်ကို တည်ဆောက်တဲ့အချိန်မှာ “ဒီလိုပဲရပါပြီ” လို့ လွယ်လွယ်ထားလိုက်တာ၊ ဘယ်သူ certificate လျှောက်နိုင်တယ်ဆိုတာကို မတိကျစွာ သတ်မှတ်ထားတာလို အမှားတွေကို ဆိုလိုပါတယ်။

ဒီ module မှာတော့ enterprise environment တွေမှာ Public Key Infrastructure (PKI) ကို AD CS နဲ့ ဘယ်လိုအသုံးပြုကြတယ်ဆိုတာကို အခြေခံကစပြီး ရှင်းပြထားပါတယ်။ PKI ဆိုတာက certificate တွေကို အသုံးပြုပြီး ယုံကြည်မှုကို တည်ဆောက်တဲ့ စနစ်တစ်ခုဖြစ်ပြီး AD CS က အဲဒီ PKI ကို Windows environment အတွက် တာဝန်ယူပေးတဲ့အရာပါ။ Organization တွေက certificate ကို ဘယ်လိုသုံးတယ်၊ authentication အတွက် ဘယ်လိုအသုံးချတယ်ဆိုတာတွေကို နားလည်အောင်ရှင်းပြပြီး နောက်ပိုင်းမှာ attacker တွေက အဲဒီ feature တွေကို ဘယ်လိုပြန်လည်အသုံးချပြီး တိုက်ခိုက်နိုင်တယ်ဆိုတာကိုလည်း ဆက်လက်ရှင်းပြထားပါတယ်။

PKI & AD CS

Active Directory Certificate Services (AD CS) ဆိုတာက Windows Server မှာ ထည့်သွင်းနိုင်တဲ့ role တစ်ခုဖြစ်ပြီး organization တစ်ခုအတွက် Public Key Infrastructure (PKI) ကို တည်ဆောက်နိုင်အောင် လုပ်ပေးတဲ့ service ပါ။ အရေးကြီးဆုံးအချက်ကတော့ ဒီ PKI ကို Active Directory နဲ့ ချိတ်ဆက်ပြီး စီမံခန့်ခွဲနိုင်တာပါ။ Active Directory နဲ့ ချိတ်ထားတဲ့အတွက် certificate ထုတ်ယူတာ၊ certificate ကို ပယ်ဖျက်တာ (revocation) လို အလုပ်တွေကို အလွန်လွယ်ကူအောင် လုပ်နိုင်ပြီး AD မှာရှိတဲ့ authentication၊ permission၊ group စတဲ့ security concept တွေကိုပဲ ပြန်လည်အသုံးချနိုင်ပါတယ်။ အဲ့

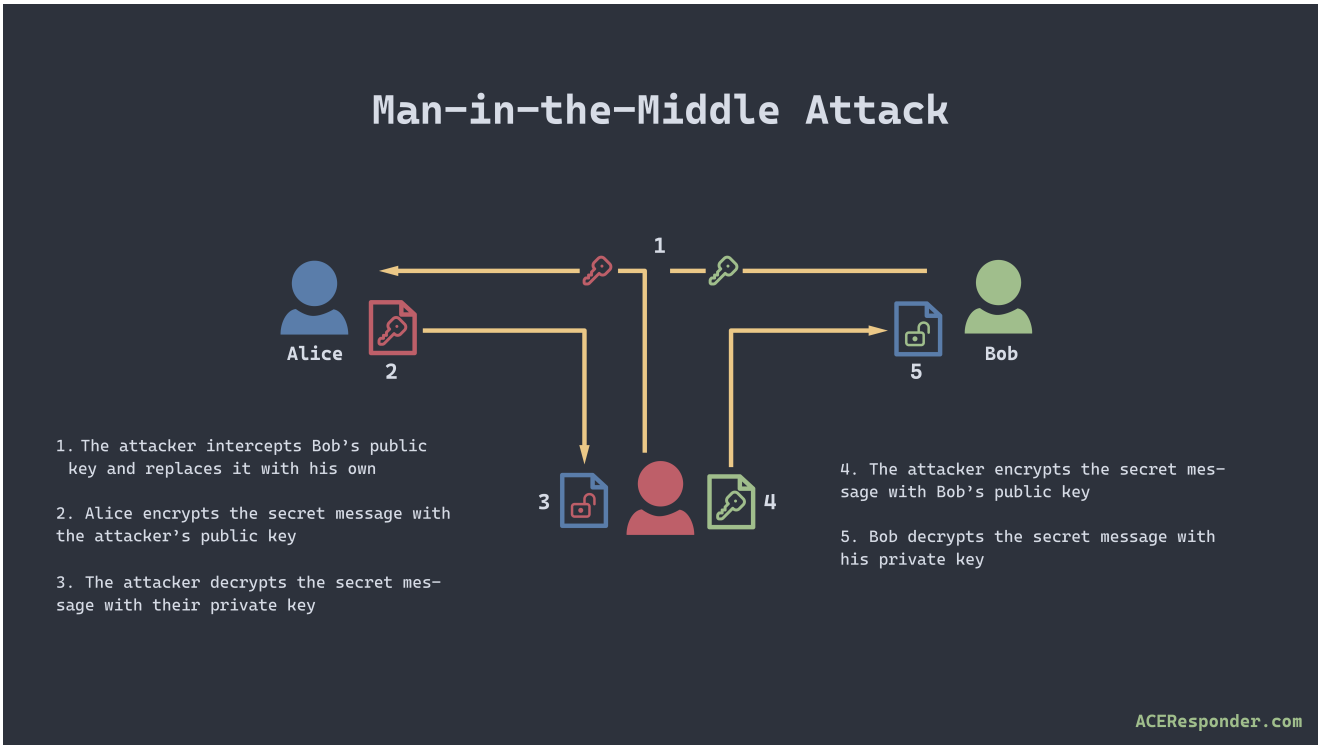
ဒါကြောင့် administrator တွေအတွက် စနစ်တကျ ထိန်းချုပ်ရလွယ်ပြီး enterprise environment တွေအတွက် အလွန်အသုံးဝင်ပါတယ်။

PKI

ဒါကို နားလည်ဖို့ အရင်ဆုံး PKI ဆိုတာဘာလဲဆိုတာကို နားလည်ရပါမယ်။ PKI ဆိုတာက လူတစ်ယောက် ဒါမှမဟုတ် system တစ်ခုရဲ့ identity ကို အတည်ပြုပေးနိုင်တဲ့ infrastructure တစ်ခုဖြစ်ပါတယ်။ အထူးသဖြင့် public key cryptography ကို အသုံးပြုပြီး “ဒီ key ဟာ ဒီသူရဲ့ key အမှန်ပဲ” ဆိုတာကို ယုံကြည်နိုင်အောင် လုပ်ပေးပါတယ်။

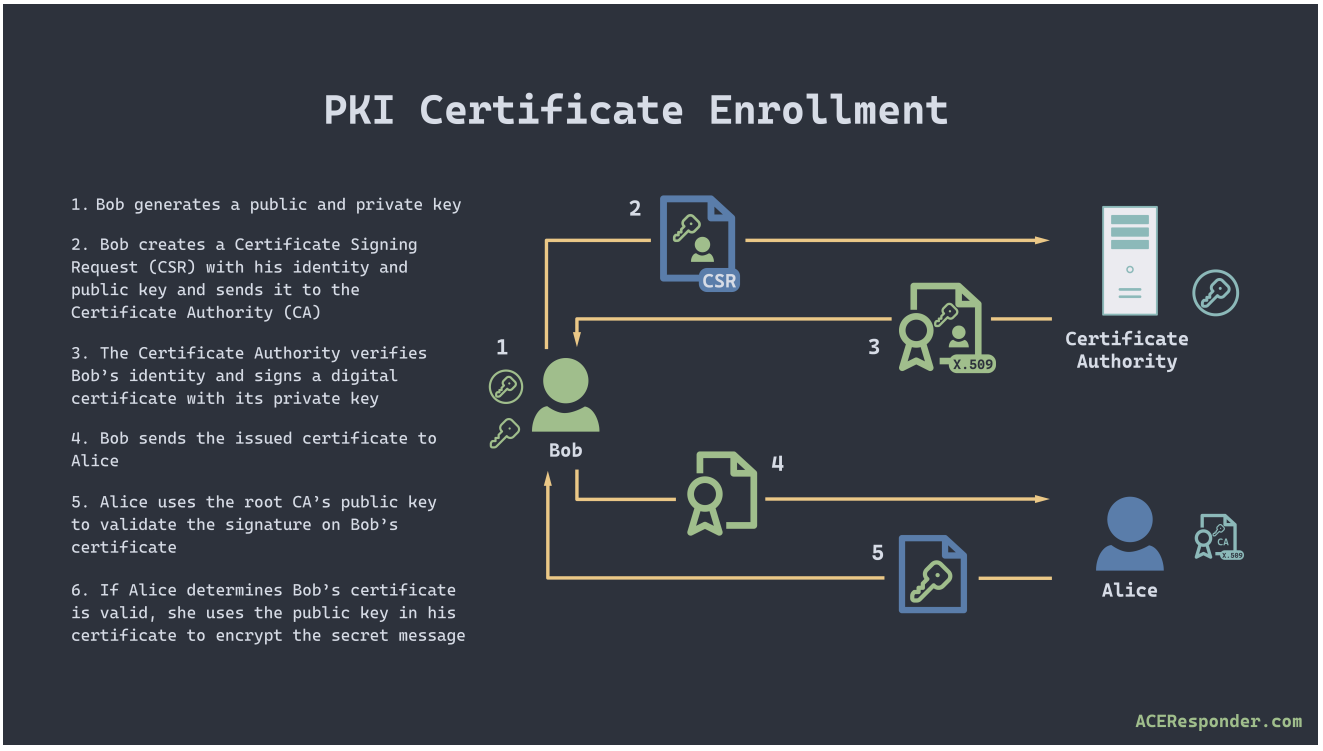
ဥပမာတစ်ခုပေးရမယ်ဆိုရင် သင်က သင့်သူငယ်ချင်း Bob ကို လျှို့ဝှက်စာတစ်စောင် ပို့ချင်တယ်လို့ စဉ်းစားပါ။ သင် message ကို encrypt လုပ်ပြီး key ကို Bob ဆီ ပို့လိုက်မယ်ဆိုရင် လမ်းမှာ တခြားလူတစ်ယောက်က key ကို ခိုးယူသွားနိုင်တဲ့ အန္တရာယ်ရှိပါတယ်။ ဒီပြဿနာကို ဖြေရှင်းဖို့ asymmetric cryptography ကို အသုံးပြုပါတယ်။ Bob က သူ့ရဲ့ public key ကို အများသိအောင် ထုတ်ပြထားမယ်ဆိုရင် သင်က အဲဒီ public key နဲ့ message ကို encrypt လုပ်နိုင်ပါတယ်။ အဲ့ဒီ message ကို Bob ရဲ့ private key နဲ့သာ decrypt လုပ်နိုင်တာကြောင့် လုံခြုံပါတယ်။

ဒါပေမယ့် ဒီနေရာမှာ အရေးကြီးတဲ့ မေးခွန်းတစ်ခု ပေါ်လာပါတယ်။ သင်ရတဲ့ public key က တကယ်ပဲ Bob ရဲ့ public key ဟုတ်မဟုတ်ကို ဘယ်လိုသိမလဲဆိုတာပါ။ Asymmetric cryptography တစ်ခုတည်းနဲ့ဆိုရင် ဒီ identity verification ပြဿနာကို မဖြေရှင်းနိုင်ပါဘူး။ Attacker တစ်ယောက်က “ဒါ Bob ရဲ့ public key ပါ” လို့ လိမ်ပြနိုင်ပါတယ်။



ဒီနေရာမှာ PKI က အရေးကြီးလာပါတယ်။ PKI က public key ရဲ့ identity ကို အတည်ပြုပေးနိုင်တဲ့ service တွေကို ပေးပါတယ်။ Bob က သူ့ရဲ့ public key ကို PKI ထဲက **Certificate Authority (CA)** ဆီ တင်ပြီး certificate တစ်ခု လျှောက်နိုင်ပါတယ်။ CA က Bob ရဲ့ identity ကို စစ်ဆေးပြီး သူ့ public key ကို အတည်ပြုပေးတဲ့ digital certificate တစ်ခု ထုတ်ပေးပါတယ်။ သင် Bob ဆီကို message ပို့ချင်တဲ့အချိန်မှာ Bob ဆီက certificate ကို တောင်းယူပြီး အဲဒီ certificate ကို အသုံးပြုပြီး Bob ရဲ့ public key ကို ယုံကြည်နိုင်ပါတယ်။

ဒီ digital certificate ရဲ့ အထူးအချက်က CA ရဲ့ private key နဲ့ sign လုပ်ထားတာပါ။ CA က certificate ထဲမှာ ပါတဲ့ အချက်အလက်တွေကို hash လုပ်ပြီး အဲဒီ hash ကို သူ့ရဲ့ private key နဲ့ encrypt လုပ်ပါတယ်။ အဲဒီ signature ကို certificate ထဲမှာ ထည့်ထားပါတယ်။ Certificate ကို လက်ခံတဲ့သူက CA ရဲ့ public key ကို အသုံးပြုပြီး signature ကို verify လုပ်နိုင်ပါတယ်။ Hash တန်ဖိုးက ကိုက်ညီတယ်ဆိုရင် **"ဒီ certificate ကို ယုံကြည်ရတဲ့ CA က ထုတ်ပေးတာပါ"** လို့ အတည်ပြုနိုင်ပြီး Bob ရဲ့ public key က လည်း အမှန်ဖြစ်ကြောင်း ယုံကြည်နိုင်ပါတယ်။ AD CS environment မှာဆိုရင် ဒီ CA က AD CS role တင်ထားတဲ့ server ပဲ ဖြစ်ပါတယ်။



Enterprise environment တစ်ခုမှာ CA က Active Directory ကို အသုံးပြုပြီး Bob ရဲ့ identity ကို စစ်ဆေးပါတယ်။ Bob က AD domain ထဲမှာ login ဝင်ထားပြီး authenticated ဖြစ်နေတာကြောင့် "ဒီ request လုပ်တဲ့သူဟာ Bob အမှန်ပဲ" လို့ CA က ယုံကြည်ပါတယ်။ ဘယ်သူ certificate လျှောက်နိုင်တယ်၊ ဘယ်လိုအသုံးပြုနိုင်တဲ့ certificate ကို ထုတ်ပေးမလဲဆိုတာတွေကို certificate template တွေနဲ့ အရင်ကတည်းက configure လုပ်ထားပါတယ်။

ဒီအခါမှာ "ဘာကြောင့် Bob ရဲ့ public key ကို မယုံဘဲ root CA ရဲ့ public key ကို ယုံကြည်တာလဲ" ဆိုတဲ့ မေးခွန်း ပေါ်လာနိုင်ပါတယ်။ အကြောင်းရင်းက certificate distribution ပုံစံကြောင့်ပါ။ Windows operating system တွေမှာ မူလကတည်းက trusted root CA certificate တွေ ပါလာပါတယ်။ AD CS ရဲ့ root CA certificate ကိုတော့ group policy နဲ့ AD domain အတွင်းက computer တွေဆီကို ဖြန့်ဝေပါတယ်။ Certificate တစ်ခုကို စစ်ဆေးတဲ့အချိန်မှာ system က certificate store ထဲက trusted root certificate တွေနဲ့ နှိုင်းယှဉ်စစ်ဆေးပါတယ်။ Root CA ကို ယုံကြည်တယ်ဆိုရင် အဲဒီ root CA နဲ့ sign လုပ်ထားတဲ့ certificate အားလုံးကိုလည်း ယုံကြည်နိုင်ပါတယ်။

ဒါပေမယ့် ဒီယုံကြည်မှုအားလုံးက "AD domain ရဲ့ integrity" ပေါ်မှာ မူတည်ပါတယ်။ AD CS ကိုလည်း အမှန်တကယ် secure ဖြစ်အောင် configure လုပ်ထားရပါမယ်။ 2021 ခုနှစ်မှာ SpecterOps က AD CS ရဲ့ security အန္တရာယ်တွေကို သုတေသနလုပ်ပြီး ထုတ်ပြန်ခဲ့ပါတယ်။ အဲဒီ research က organization အများစုကို ထိတ်လန့်စေခဲ့ပါတယ်။ Whitepaper ထဲမှာ persistence ရယူနိုင်တဲ့ attack တွေ၊ certificate ခိုးယူတဲ့ နည်းလမ်းတွေ၊ AD forest တစ်ခုကနေ တစ်ခုကို ကူးသန်းနိုင်တဲ့ attack တွေကို ဖော်ပြထားပါတယ်။

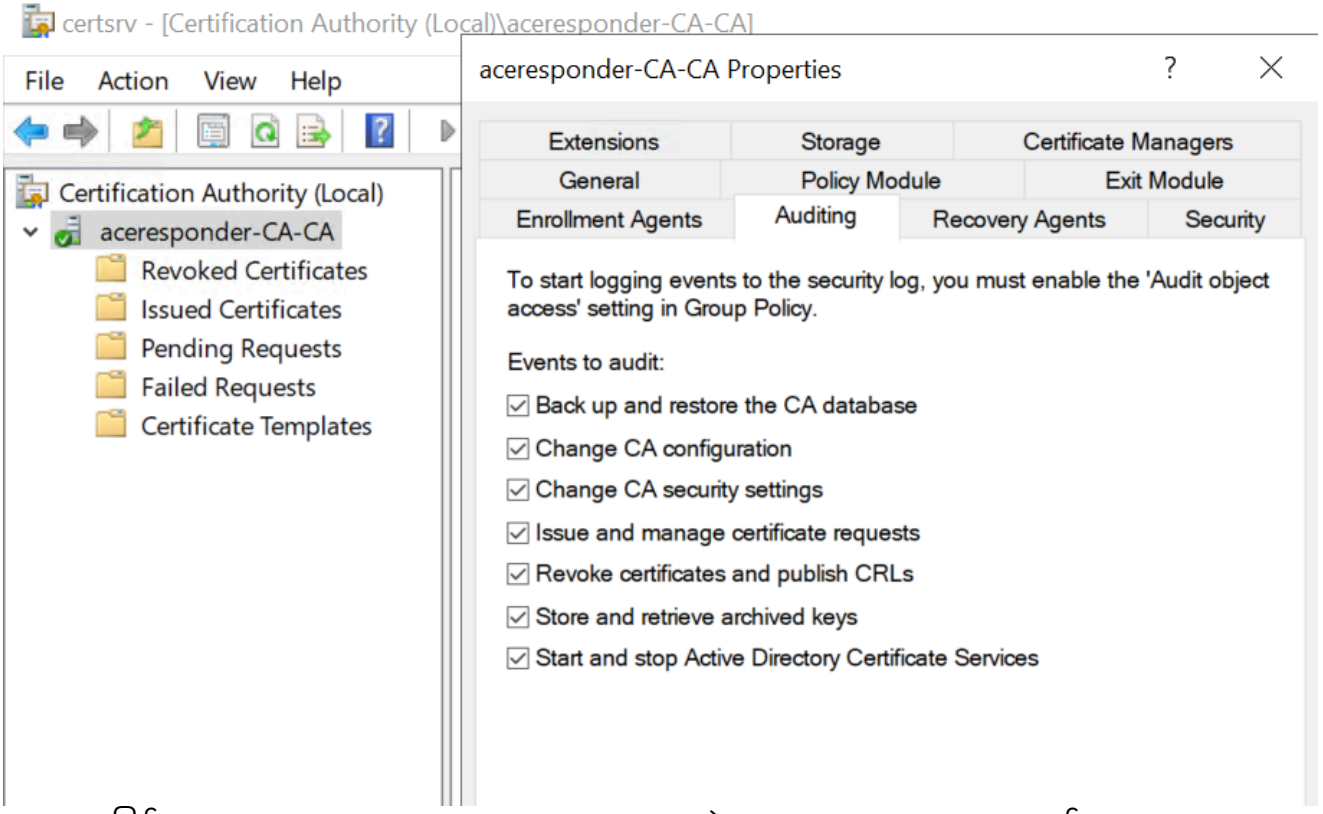
ဒါပေမယ့် အထိရောက်ဆုံး အန္တရာယ်က certificate enrollment process ကို abuse လုပ်ပြီး privilege escalation လုပ်နိုင်တာပါ။ Certificate template တွေကို misconfiguration လုပ်ထားတဲ့အခါ low-privilege user တစ်ယောက်က high-value account တွေအတွက် certificate ထုတ်ယူနိုင်တဲ့ အခြေအနေတွေ ရှိနေတတ်ပါတယ်။ ဒီလို misconfiguration တွေဟာ အလွန်အများအပြား network တွေမှာ တွေ့ရပြီး attacker အတွက် one-step နဲ့ system ကို pwn လုပ်နိုင်တဲ့ အခွင့်အရေး ဖြစ်လာပါတယ်။

AD CS Telemetry

AD CS Telemetry ဆိုတာက Active Directory Certificate Services ကို အသုံးပြုနေတဲ့အချိန်မှာ **ဘာတွေဖြစ်နေတယ်၊ ဘယ်သူက certificate လျှောက်ထားတယ်၊ ဘယ်လိုပုံစံနဲ့ လျှောက်ထားတယ်ဆိုတာတွေကို log အနေနဲ့ မှတ်တမ်းတင်ထားတဲ့ အချက်အလက်**တွေကို ဆိုလိုပါတယ်။ ဒီ telemetry ကို သုံးပြီး administrator ဒါမှမဟုတ် defender တွေက မမှန်ကန်တဲ့ certificate request တွေ၊ attacker ရဲ့ လှုပ်ရှားမှုတွေကို စုံစမ်းစစ်ဆေးနိုင်ပါတယ်။

AD CS နဲ့ ဆိုင်တဲ့ ပုံမှန် event log တွေကို Windows ရဲ့ Security Log ထဲမှာ တွေ့နိုင်ပါတယ်။ ဒါပေမယ့် အရေးကြီးတဲ့အချက်က ဒီ log တွေဟာ default အနေနဲ့ ဖွင့်ထားတာ မဟုတ်ပါဘူး။ Administrator က Audit Policy ကို ကိုယ်တိုင် configure လုပ်မှသာ event တွေကို စတင်မှတ်တမ်းတင်ပါတယ်။

To enable : `Advanced Audit Policy Configuration > Object Access`



အထူးသဖြင့် Advanced Audit Policy Configuration ထဲက Object Access အောက်မှာ Audit Certification Services ကို enable လုပ်ရပါမယ်။ ဒီ audit ကို ဖွင့်လိုက်ရင် **CA server မှာ certificate request၊ issuance၊ failure စတဲ့ event အမျိုးမျိုးကို log အဖြစ် သိမ်းဆည်းပေးပါတယ်။** The comprehensive list of events this enables can be found here. ဒီ event တွေကို CA server properties နဲ့ ဆက်နွယ်နေတဲ့ log တွေလို့လည်း ပြောနိုင်ပါတယ်။

ဒါပေမယ့် လက်တွေ့မှာ AD CS telemetry ဟာ အမြဲတမ်း ပြည့်စုံတိကျနေတယ်လို့ မဆိုနိုင်ပါဘူး။ အကြောင်းက AD CS server process ဖြစ်တဲ့ `certsrv.exe` က certificate request ကို လက်ခံတဲ့ နည်းလမ်း နှစ်မျိုးရှိလို့ပါ။ ဒီနည်းလမ်းနှစ်မျိုးက protocol မတူဘဲ code path မတူတာကြောင့် log ထဲကို ထည့်ပေးတဲ့ အချက်အလက်တွေလည်း ကွာခြားသွားပါတယ်။

ပထမနည်းလမ်းကတော့ RPC ကို အသုံးပြုပြီး ICertPassage protocol (MS-ICPR)နဲ့ certificate request ပို့တဲ့ ပုံမှန်နည်းလမ်းပါ။ ဒီနည်းလမ်းနဲ့ request လုပ်လာတဲ့အခါ Windows က certificate request ထဲက အရေးကြီးတဲ့ attribute တွေကို extract လုပ်ပြီး log ထဲမှာ သိမ်းပေးပါတယ်။ ဒီ attribute တွေထဲမှာ ဘယ် certificate template ကို သုံးထားလဲ၊ Subject Alternative Name (SAN) ဘာတွေပါလဲဆိုတာတွေ ပါဝင်ပါတယ်။ Defender တစ်ယောက်အနေနဲ့ ဒီအချက်အလက်တွေကို ကြည့်ပြီး “ဒီ template ကို abuse လုပ်ထားလား” ဆိုတာကို အကဲဖြတ်နိုင်ပါတယ်။

ဒုတိယနည်းလမ်းကတော့ object-based ဖြစ်တဲ့ DCOM ကို အသုံးပြုပြီး Windows Client Certificate Enrollment protocol(MS-WCCE) နဲ့ request လုပ်တဲ့နည်းလမ်းပါ။ ဒီနည်းလမ်းနဲ့ request လုပ်လာတဲ့အခါ certificate request attribute တွေကို log ထဲမှာ မပြည့်စုံအောင်ပဲ ထည့်ပေးပါတယ်။ ဆိုလိုတာက template အမည်၊ SAN လို အရေးကြီးတဲ့ အချက်အလက်တွေကို log ထဲမှာ မမြင်ရနိုင်ပါဘူး။ အဲ့ဒါကြောင့် forensic investigation လုပ်တဲ့အခါ အလွန်ခက်ခဲသွားပါတယ်။

ဒီအချက်ကို attacker tooling နဲ့ ချိတ်ဆက်ကြည့်မယ်ဆိုရင် ပိုပြီး ရှင်းလင်းလာပါတယ်။ Attacker က Certipy ကို အသုံးပြုပြီး certificate request လုပ်တယ်ဆိုရင် အများအားဖြင့် RPC (MS-ICPR) နည်းလမ်းကို သုံးပါတယ်။ ဒီလိုဆိုရင် native AD CS logs ထဲမှာ attribute တွေပါလာတာကြောင့် defender က စုံစမ်းစစ်ဆေးနိုင်ပါသေးတယ်။ ဒါပေမယ့် C# နဲ့ရေးထားတဲ့ Certify tool ကို အသုံးပြုတဲ့အခါ DCOM interface ကို သုံးတာဖြစ်ပြီး certificate template၊ SAN စတဲ့ အရေးကြီးတဲ့ attribute တွေကို log ထဲမှာ မမြင်ရတော့ပါဘူး။ ဒီအခြေအနေမှာ attacker ရဲ့လှုပ်ရှားမှုကို native telemetry တစ်ခုတည်းနဲ့ ဖော်ထုတ်ဖို့ အလွန်ခက်ခဲသွားပါတယ်။

ဒီပြဿနာကို ဖြေရှင်းဖို့ SpecterOps တို့က RPC Firewall ကို ပြင်ဆင်ပြီး certificate request ထဲက အချက်အလက်ပိုများစွာကို extract လုပ်နိုင်အောင် ပြုလုပ်ထားပါတယ်။ ဒီလိုပြင်ဆင်ထားတဲ့ RPC Firewall ကနေ ထွက်လာတဲ့ event တွေကို WinlogProviderName ကို RPCFW လို့ထားပြီး event code 3 အနေနဲ့ တွေ့နိုင်ပါတယ်။

> event code 3 in `WinlogProviderName == "RPCFW"`

ဒီ event တွေက DCOM နည်းလမ်းကို သုံးထားတောင် certificate request အကြောင်းကို ပိုမိုမြင်နိုင်အောင် ကူညီပေးပါတယ်။

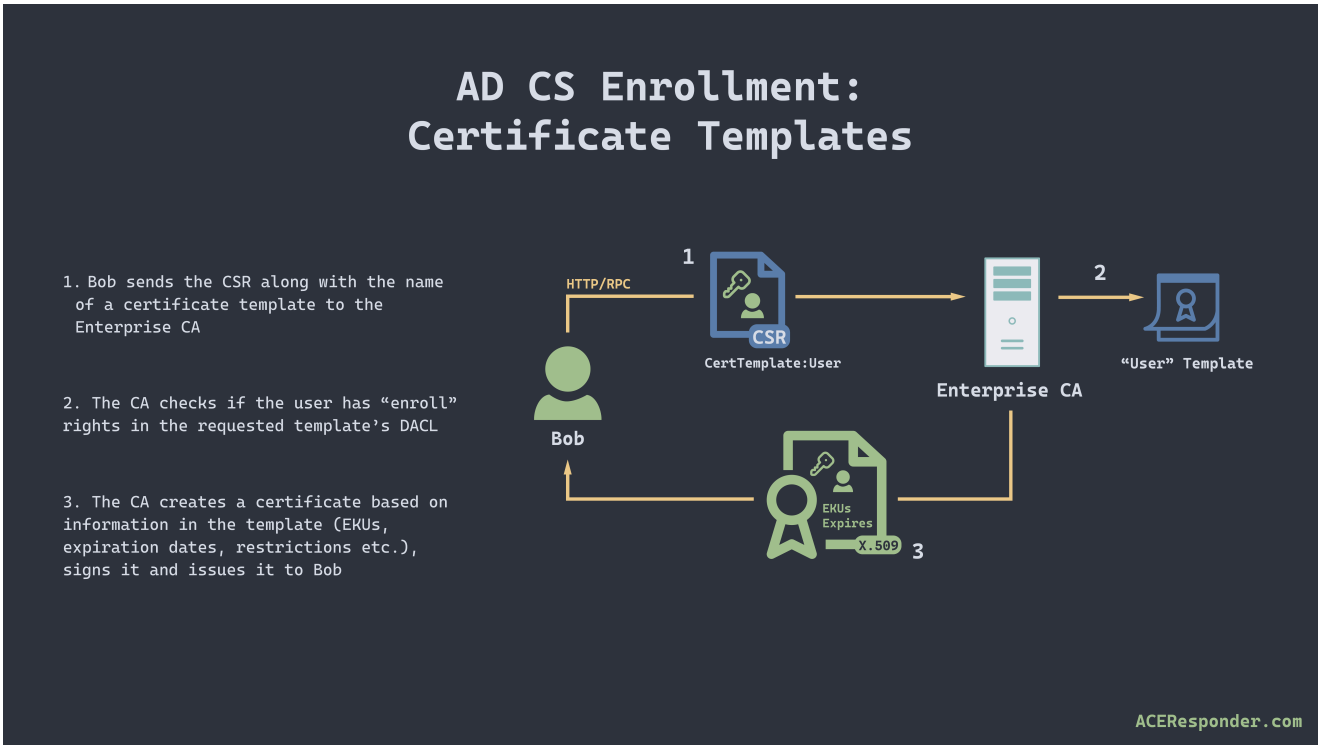
Certificate Template Misconfigurations

Certificate template ဆိုတာက AD CS ထဲမှာ certificate လျှောက်တဲ့ လုပ်ငန်းစဉ်ကို အလိုအလျောက် စနစ်တကျလုပ်ပေးဖို့ အသုံးပြုတဲ့ configuration တစ်ခုပါ။ User တစ်ယောက်ဖြစ်တဲ့ Bob က certificate လျှောက်တဲ့အခါ CA ဆီကို Certificate Signing Request (CSR) တစ်ခု ပို့ပါတယ်။ ဒီ CSR ထဲမှာ Bob ရဲ့ public key နဲ့ “ဒီလို certificate လိုချင်ပါတယ်” ဆိုတဲ့ request အချက်အလက်တွေ ပါဝင်ပါတယ်။ CSR နဲ့အတူ Bob က ဘယ် certificate template ကို အသုံးပြုမယ်ဆိုတာကိုလည်း CA ကို ပြောရပါတယ်။ CA

က certificate ထုတ်ပေးမယ်ဆိုရင် အရင်ဆုံး ဒီ CSR ကို သတ်မှတ်ထားတဲ့ template နဲ့ နှိုင်းယှဉ်စစ်ဆေးပါတယ်။

Certificate template က CA ကို "ဒီ certificate ကို ဘယ်လိုပုံစံနဲ့ ထုတ်ပေးရမလဲ" ဆိုတာကို လမ်းညွှန်ပေးပါတယ်။ Template ထဲမှာ ဘယ် user သို့မဟုတ် group တွေက ဒီ template နဲ့ certificate လျှောက်ခွင့်ရှိတယ်ဆိုတာ သတ်မှတ်ထားပါတယ်။ ဒါ့အပြင် CSR ထဲက ဘယ် field တွေကို ခွင့်ပြုမလဲ၊ ဘယ် value တွေကို လက်ခံမလဲဆိုတာကိုလည်း template က ဆုံးဖြတ်ပေးပါတယ်။ Certificate ထုတ်ပြီးသွားရင် အဲ့ဒီ certificate ကို ဘာအတွက် သုံးခွင့်ရှိမလဲ၊ ဥပမာ authentication အတွက်လား၊ code signing အတွက်လား၊ email encryption အတွက်လား ဆိုတာတွေကိုပါ သတ်မှတ်ထားပါတယ်။ နောက်ဆုံးအနေနဲ့ certificate ရဲ့ သက်တမ်းဘယ်လောက်ရှိမလဲဆိုတာကိုလည်း template က ဆုံးဖြတ်ပါတယ်။

CA က template ရဲ့ စည်းမျဉ်းအားလုံးနဲ့ Bob ရဲ့ CSR ကို စစ်ဆေးပြီး အားလုံးကိုက်ညီတယ်ဆိုရင် certificate တစ်ခုကို ဖန်တီးပါတယ်။ ဒီ certificate ထဲမှာ template ထဲက setting တွေနဲ့ Bob ရဲ့ request အချက်အလက်တွေကို ပေါင်းစပ်ထားပါတယ်။ Certificate ကို Bob ဆီ ပြန်ပို့ပေးပြီး CA က သူထုတ်ပေးထားတဲ့ certificate ကို သူ့ရဲ့ database ထဲမှာလည်း သိမ်းထားပါတယ်။ ဒါကြောင့် နောက်ပိုင်း forensic စစ်ဆေးတဲ့အခါ CA server ပေါ်မှာ issued certificate တွေကို ပြန်ကြည့်နိုင်ပါတယ်။



Certificate template ရဲ့ အရေးကြီးတဲ့ အစိတ်အပိုင်းတစ်ခုက EKU (Extended Key Usage) ပါ။ EKU ဆိုတာက ဒီ certificate ကို ဘာလုပ်ဖို့ သုံးခွင့်ရှိတယ်ဆိုတာကို သတ်မှတ်ပေးတဲ့ အချက်အလက်ဖြစ်ပါတယ်။ ဥပမာ Bob နဲ့ signed and encrypted email ပို့ချင်တယ်ဆိုရင် သူ့ certificate ထဲမှာ Secure Email EKU ပါဝင်ရပါမယ်။ EKU မတူတာနဲ့ certificate ရဲ့ အသုံးချနိုင်တဲ့ အလုပ်တွေ မတူတော့ပါဘူး။ တချို့ EKU တွေက code signing လုပ်ခွင့်ပေးပြီး တချို့က authentication လုပ်ခွင့်ပေးပါတယ်။ ဒီလို EKU မျိုးစုံရှိတာကြောင့် လိုအပ်တဲ့ အလုပ်အတွက် ကိုက်ညီတဲ့ template ကို ရွေးပြီး certificate လျှောက်နိုင်တာဟာ template တွေရဲ့ အားသာချက်ပါ။

ဒီနေရာမှာ အလွန်အရေးကြီးတဲ့ သဘောတရားတစ်ခုရှိပါတယ်။ Certificate နဲ့ အဲ့ဒီ certificate ရဲ့ private key ကို ကိုင်ထားနိုင်ပြီဆိုရင် အဲ့ဒီ identity ကို ကိုယ်စားပြုနိုင်သွားပါတယ်။ ထပ်ပြီး password မလိုတော့ပါဘူး။ Attacker တစ်ယောက်က Bob ရဲ့ private key ကို ခိုးယူနိုင်မယ်ဆိုရင် certificate ထဲက EKU ခွင့်ပြုထားသလောက်ကို Bob အနေနဲ့ လုပ်နိုင်သွားပါတယ်။ အထူးသဖြင့် certificate ထဲမှာ authentication

EKU ပါဝင်နေရင် attacker က Bob ရဲ့ password ကို မသိဘဲ AD domain ထဲကို Bob အနေနဲ့ login ဝင်နိုင်ပါတယ်။

Attacker က Bob ရဲ့ private key ကို ခိုးယူတာမလုပ်ဘဲလည်း Bob အဖြစ် certificate လျှောက်ယူနိုင်တဲ့ နည်းလမ်းတွေရှိပါတယ်။ အလွယ်ဆုံးနည်းလမ်းက Bob ရဲ့ account ကို compromise လုပ်ပြီး Bob အနေနဲ့ certificate လျှောက်လိုက်တာပါ။ Attacker က ကိုယ်ပိုင် private-public key pair ကို generate လုပ်ပြီး CA ကို "ဒီ public key ကို Bob အဖြစ် sign လုပ်ပေးပါ" လို့ လျှောက်နိုင်ပါတယ်။ ဒီလိုဆိုရင် CA က template အတိုင်း စစ်ဆေးပြီး အမှားမတွေ့ရင် Bob ရဲ့ identity ပါတဲ့ certificate ကို ထုတ်ပေးသွားနိုင်ပါတယ်။

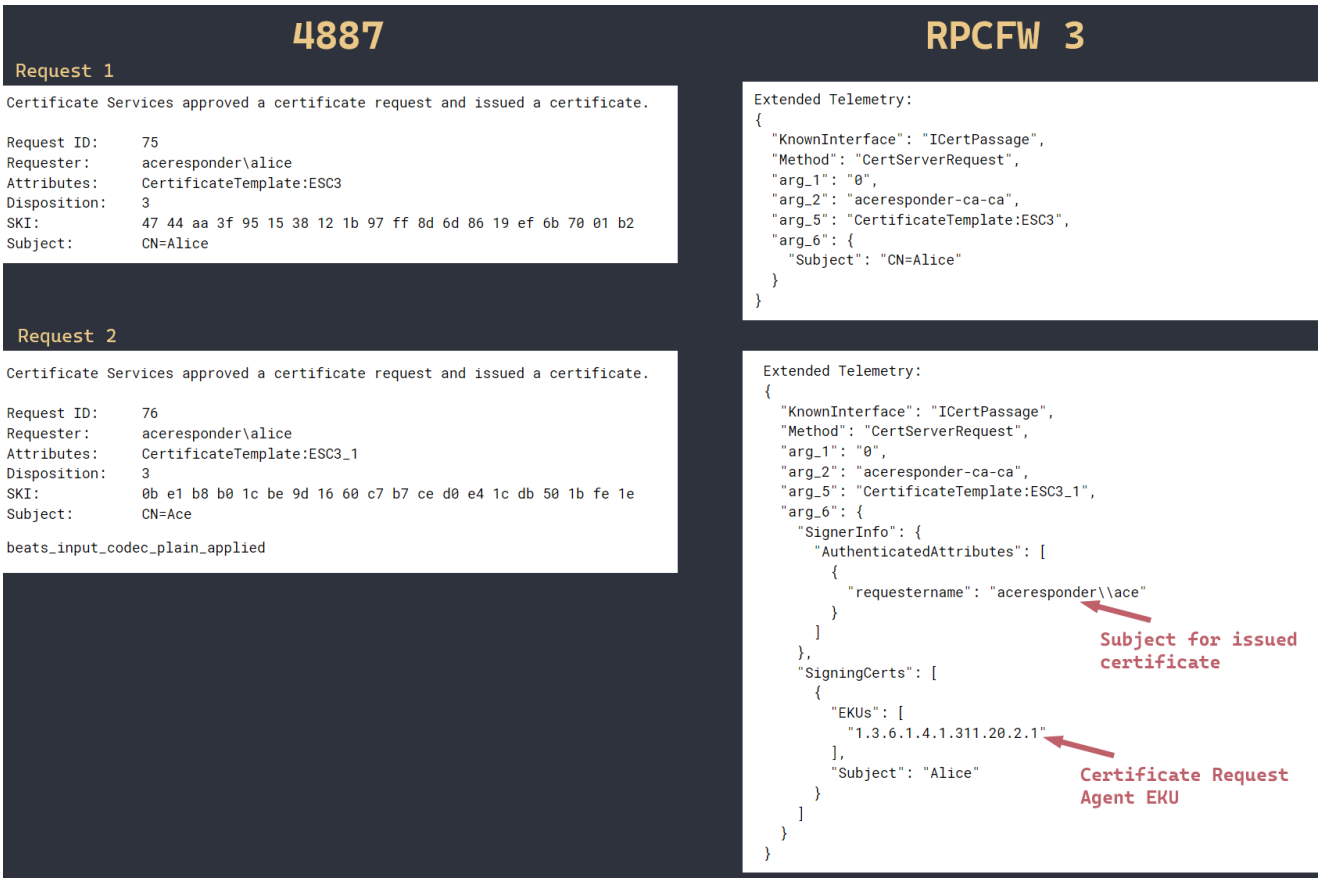
နောက်ထပ် အန္တရာယ်ကြီးတဲ့ နည်းလမ်းတစ်ခုက certificate template တွေက Subject Alternative Name (SAN) ကို requestor က ကိုယ်တိုင် သတ်မှတ်ခွင့်ပေးထားတဲ့အခါ ဖြစ်ပါတယ်။ SAN ဆိုတာက certificate တစ်ခုထဲမှာ identity တစ်ခုထက်ပိုထည့်နိုင်အောင်လုပ်ပေးတဲ့ feature ပါ။ ဥပမာ website တစ်ခုက TLS certificate တစ်ခုတည်းနဲ့ subdomain အများကြီးကို အသုံးပြုနိုင်တာက SAN ကြောင့်ပါ။ Browser က certificate ကို စစ်တဲ့အခါ Server Authentication EKU ရှိလား၊ domain name က certificate ထဲက Subject ဒါမှမဟုတ် SAN နဲ့ ကိုက်ညီလားဆိုတာကို စစ်ဆေးပါတယ်။

အလားတူပဲ AD environment ထဲမှာ authentication EKU ပါတဲ့ certificate template တစ်ခုက low-privilege user တွေကို SAN ကို ကိုယ်တိုင်သတ်မှတ်ခွင့်ပေးထားမယ်ဆိုရင် attacker က SAN ထဲမှာ ဘယ် user name မဆို ထည့်နိုင်ပါတယ်။ အဲ့ဒီအခါ AD domain ထဲက service တွေက SAN ထဲက alternate identity ကို ယုံကြည်ပြီး authentication လက်ခံပေးသွားပါတယ်။ Browser က website certificate ထဲက subdomain ကို ယုံကြည်သလိုပဲ AD က SAN ထဲက principal ကို ယုံကြည်တာ ဖြစ်ပါတယ်။

ဒီလို vulnerability ကို **ESC1** လို့ အမည်ပေးထားပါတယ်။ ESC1 ဖြစ်ဖို့အတွက် condition အများကြီးရှိပေမယ့် အဓိကအချက်ကတော့

- low-privilege user တွေကို enrollment လုပ်ခွင့်ပေးထားတာ၊
- authentication EKU ပါဝင်နေတာနဲ့ request လုပ်တဲ့သူက SAN ကို ကိုယ်တိုင် သတ်မှတ်နိုင်တာပဲ ဖြစ်ပါတယ်။
- ဒီ SAN ကို သတ်မှတ်နိုင်စေတဲ့ template setting က Subject Name ကို "Supply in the request" လို့ထားထားတာပါ။
  နာမည်ကြည့်ရင်ပဲ လုံခြုံရေးအန္တရာယ်ကြီးတယ်လို့ ထင်ရပေမယ့် SpecterOps က research ထုတ်မချင်း ဒီအချက်ကို အလေးထားပြီး စစ်ဆေးသူတွေ များမဟုတ်ပါဘူး။ PKI ကို စနစ်တကျ စီမံခန့်ခွဲရတာ အလွန်ခက်ခဲတဲ့အတွက် troubleshooting လုပ်ရင်း ဒီလို misconfiguration တွေ မသိမသာ ဖြစ်လာတာတွေ များပါတယ်။

ESC1 နဲ့ ထုတ်ထားတဲ့ certificate တွေကို CA server ပေါ်က Issued Certificates ထဲမှာ ကြည့်နိုင်ပါတယ်။ Certificate ထဲမှာ Subject နဲ့ SAN နှစ်ခုစလုံး ပါဝင်နေတာကြောင့် forensic စစ်ဆေးတဲ့အခါ အလွယ်တကူ တွေ့နိုင်ပါတယ်။ ဥပမာ attacker က Alice ရဲ့ account ကို အသုံးပြုပြီး certificate လျှောက်ထားပေမယ့် SAN ထဲမှာ domain Administrator account ကို ထည့်ထားတဲ့အခြေအနေကို CA database ထဲမှာ တိုက်ရိုက်မြင်နိုင်ပါတယ်။



Log တွေထဲမှာလည်း ESC1 ရဲ့ လက္ခဏာတွေကို တွေ့နိုင်ပါတယ်။ Certificate request နဲ့ issuance event တွေဟာ သီးခြား event အနေနဲ့ ပေါ်လာပေမယ့် ပုံမှန်အားဖြင့် Request ID ကို အသုံးပြုပြီး certsrv database ထဲက certificate နဲ့ ဆက်စပ်စစ်ဆေးနိုင်ပါတယ်။ ဒီလိုနားလည်ထားခြင်းက AD CS abuse ကို detect လုပ်ရာမှာ အခြေခံအလွန်အရေးကြီးတဲ့ အဆင့်တစ်ခု ဖြစ်ပါတယ်။

Test : ESC1

Which user, other than `ace` was the victim of an ESC1 attack?

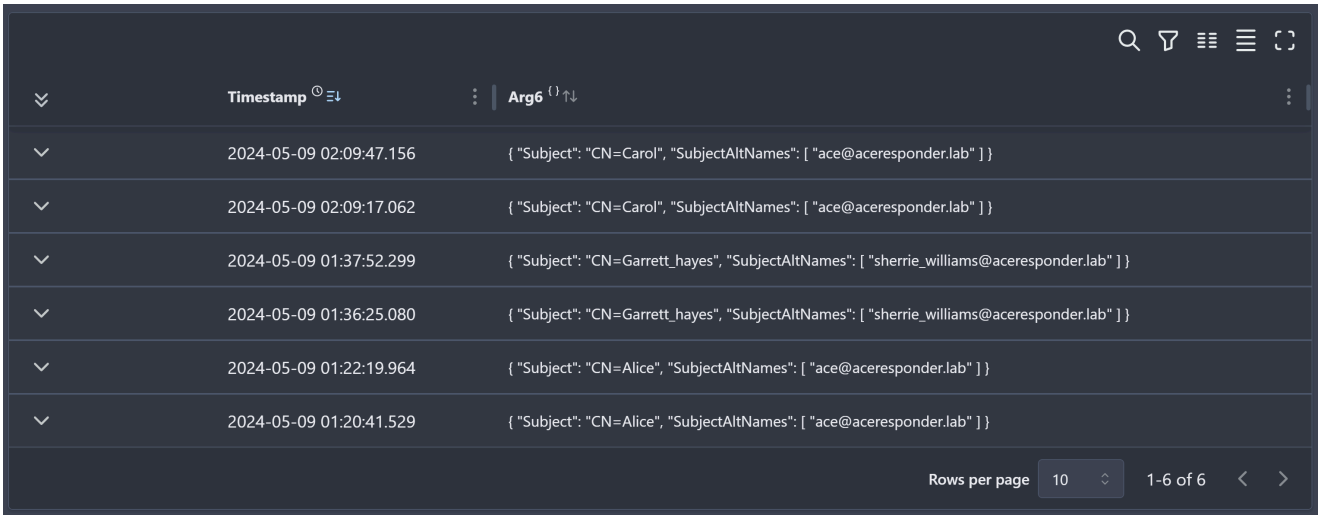
ESC1 attack ဆိုတာကို ပြန်လည်အကျဉ်းချုပ်ပြောရမယ်ဆိုရင် low-privilege user တစ်ယောက်က authentication EKU ပါတဲ့ certificate template ကို အသုံးပြုပြီး Subject Alternative Name (SAN) ကို ကိုယ်တိုင် သတ်မှတ်နိုင်တဲ့ misconfiguration ကို abuse လုပ်တာပါ။ ဒီလိုလုပ်နိုင်တဲ့အခါ attacker က ကိုယ်မဟုတ်တဲ့ တခြား user တစ်ယောက်အဖြစ် certificate ထုတ်ယူနိုင်ပြီး အဲ့ဒီ user အနေနဲ့ AD domain ထဲကို authentication ဝင်နိုင်သွားပါတယ်။ အဲ့ဒီ SAN ထဲမှာ ထည့်ခံရတဲ့ user က ESC1 attack ရဲ့ victim ဖြစ်ပါတယ်။

“ace” အပြင် ဘယ် user က ESC1 attack ရဲ့ victim ဖြစ်ခဲ့လဲဆိုတာကို သိချင်ရင် attacker က certificate request လုပ်တဲ့အချိန် SAN ကို ဘယ်လိုသတ်မှတ်ထားလဲဆိုတာကို log တွေထဲကနေ ရှာဖို့ လိုပါတယ်။ ဒီနေရာမှာ RPC Firewall (RPCFW) telemetry ကို အသုံးချနိုင်ပါတယ်။ RPCFW ကို ပြင်ဆင်ထားတဲ့ environment မှာ certificate request တွေရဲ့ အသေးစိတ်အချက်အလက်တွေကို event အနေနဲ့ သိမ်းထားပါတယ်။

```
external_table('Winlog')
| where WinlogProviderName == "RPCFW" and EventCode == 3 and
```

```
isnotempty(Arg6.SubjectAltNames)
| project Timestamp, Arg6
```

ပထမ query မှာ Winlog table ထဲက RPCFW provider ကို အသုံးပြုပြီး EventCode 3 ဖြစ်တဲ့ event တွေကို စစ်ပါတယ်။ EventCode 3 ဆိုတာက RPCFW က certificate request ကို တွေ့ရှိပြီး အသေးစိတ် extract လုပ်နိုင်ခဲ့တဲ့ အခြေအနေကို ကိုယ်စားပြုပါတယ်။ အဲ့ဒီ event ထဲမှာ Arg6.SubjectAltNames ဆိုတဲ့ field ကို စစ်ကြည့်ပြီး empty မဟုတ်တဲ့ event တွေကို ရွေးထုတ်ပါတယ်။ SubjectAltNames field ထဲမှာ ဘယ် user name ဒါမှမဟုတ် principal ကို SAN အနေနဲ့ ထည့်ထားလဲဆိုတာ ပါလာပါတယ်။ အဲ့ဒီ SAN ထဲမှာ ပါလာတဲ့ user တွေက ESC1 attack ရဲ့ victim ဖြစ်နိုင်တဲ့ user တွေပဲ ဖြစ်ပါတယ်။ ဒီလိုနည်းနဲ့ “ace” အပြင် တခြား user တစ်ယောက်ကိုလည်း SAN အနေနဲ့ ထည့်ထားခဲ့မယ်ဆိုရင် log ထဲမှာ တိုက်ရိုက်မြင်နိုင်ပါတယ်။



Timestamp	Arg6
2024-05-09 02:09:47.156	{ "Subject": "CN=Carol", "SubjectAltNames": ["ace@aceresponder.lab"] }
2024-05-09 02:09:17.062	{ "Subject": "CN=Carol", "SubjectAltNames": ["ace@aceresponder.lab"] }
2024-05-09 01:37:52.299	{ "Subject": "CN=Garrett_hayes", "SubjectAltNames": ["sherrie_williams@aceresponder.lab"] }
2024-05-09 01:36:25.080	{ "Subject": "CN=Garrett_hayes", "SubjectAltNames": ["sherrie_williams@aceresponder.lab"] }
2024-05-09 01:22:19.964	{ "Subject": "CN=Alice", "SubjectAltNames": ["ace@aceresponder.lab"] }
2024-05-09 01:20:41.529	{ "Subject": "CN=Alice", "SubjectAltNames": ["ace@aceresponder.lab"] }

Rows per page 10 1-6 of 6

Environment တစ်ခုချင်းစီအလိုက် ဒီလို detection လုပ်ရတာ လွယ်လည်း လွယ်နိုင်သလို ခက်လည်း ခက်နိုင်ပါတယ်။ အကောင်းဆုံးအခြေအနေကတော့ authentication EKU ပါတဲ့ certificate template ကို low-privilege user တွေ enrollment လုပ်နိုင်အောင် မဖွင့်ထားတာပါ။ ဒါပေမယ့် လက်တွေ့မှာ troubleshooting လုပ်ရင်း template တစ်ခုကို အချိန်အတိုအတွင်း open လုပ်ထားတာမျိုး ဖြစ်နိုင်ပြီး attacker က အဲ့ဒီအချိန်ကို အသုံးချနိုင်ပါတယ်။ အဲ့ဒီလို template “ပေါ်လာတတ်” တာကြောင့် telemetry နဲ့ စောင့်ကြည့်နေရတာ အရေးကြီးပါတယ်။

RPCFW မရှိတဲ့ environment မှာဆိုရင် conventional AD CS telemetry ကို အသုံးပြုပြီးလည်း ESC1 လက္ခဏာတွေကို ရှာနိုင်ပါတယ်။ EventCode 4887 ဆိုတာက certificate request အောင်မြင်စွာ ဖြစ်သွားတဲ့ event ဖြစ်ပြီး Attributes field ထဲမှာ request ရဲ့ အချက်အလက်တွေ ပါဝင်ပါတယ်။ အဲ့ဒီ Attributes ထဲမှာ “SAN” ဆိုတဲ့ စာသား ပါနေတယ်ဆိုရင် Subject Alternative Name ကို အသုံးပြုထားတဲ့ request ဖြစ်ပါတယ်။ ဒီလို event တွေကို စုစည်းကြည့်ရင် SAN ကို abuse လုပ်ထားတဲ့ certificate request တွေကို တွေ့နိုင်ပါတယ်။

```
external_table('Winlog')
| where EventCode == 4887 and Attributes contains "SAN"
| project Timestamp, Attributes
```

ဒါပေမယ့် ဒီနည်းလမ်းမှာ အရေးကြီးတဲ့ ကန့်သတ်ချက်တစ်ခုရှိပါတယ်။ EventCode 4887 နဲ့ SAN ကို တွေ့နိုင်တာက attacker က non-object RPC interface ဖြစ်တဲ့ ICertPassage (MS-ICPR) ကို အသုံးပြုတဲ့အခါပဲ ဖြစ်ပါတယ်။ Certipy ရဲ့ default behavior က ဒီ interface ကို သုံးတာကြောင့် Certipy

ကို သုံးတဲ့ attacker ဆိုရင် native AD CS logs နဲ့ပဲ investigation လုပ်နိုင်ပါတယ်။ ဒီ interface ကို သုံးတဲ့အခါ CA server ပေါ်မှာ `\cert` named pipe ကို ချိတ်ဆက်တဲ့ log တွေလည်း ထပ်ပေါ်တတ်ပါတယ်။

Certificate-Based Authentication

Certificate-Based Authentication ဆိုတာက username နဲ့ password မသုံးဘဲ digital certificate ကို အသုံးပြုပြီး user တစ်ယောက်ရဲ့ identity ကို အတည်ပြုတဲ့ authentication နည်းလမ်းပါ။ AD environment ထဲမှာ certificate ထဲမှာ authentication EKU ပါဝင်နေရင် အဲ့ဒီ certificate ကို အသုံးပြုပြီး Kerberos authentication လုပ်နိုင်ပါတယ်။ ဒါဟာ smart card logon၊ PKINIT လို နည်းလမ်းတွေမှာ အခြေခံအဖြစ် အသုံးပြုထားတာ ဖြစ်ပါတယ်။

Attacker တစ်ယောက်က authentication EKU ပါတဲ့ certificate တစ်ခုကို ရရှိသွားမယ်ဆိုရင် password မလိုဘဲ AD domain ထဲကို ဝင်နိုင်တဲ့ အခြေအနေ ဖြစ်လာပါတယ်။ Attacker က Kerberos ရဲ့ Ticket Granting Ticket (TGT) ကို request လုပ်ပါတယ်။ TGT ဆိုတာက Kerberos authentication ရဲ့ ပထမ အဆင့်ဖြစ်ပြီး “ဒီ user ကို ယုံကြည်ပါတယ်” ဆိုတဲ့ အတည်ပြုချက်လိုပါပဲ။ TGT ရရှိပြီးသွားရင် attacker က domain ထဲက service အမျိုးမျိုးကို ဆက်လက်အသုံးပြုနိုင်ပါတယ်။

Certificate ကို အသုံးပြုပြီး TGT ကို request လုပ်တဲ့အချိန်မှာ Domain Controller ပေါ်မှာ security log event တစ်ခု ထွက်ပေါ်လာပါတယ်။ ဒီ event က **Event ID 4768** ဖြစ်ပြီး Kerberos authentication စတင်တဲ့အခါ မှတ်တမ်းတင်တဲ့ event ပါ။ Certificate-based authentication ဖြစ်တဲ့အခါ ဒီ event ထဲမှာ အသုံးပြုထားတဲ့ certificate ရဲ့ serial number နဲ့ thumbprint ကို မှတ်တမ်းတင်ထားပါတယ်။ Thumbprint ဆိုတာက certificate ကို ကိုယ်စားပြုတဲ့ hash တန်ဖိုးတစ်ခုလိုပါပဲ။

```
A Kerberos authentication ticket (TGT) was requested.

Account Information:
        Account Name:           ace
        Supplied Realm Name:    ACERESPONDER.LAB
        User ID:                        S-1-5-21-1553278193-1769982261-2954363648-500

Service Information:
        Service Name:           krbtgt
        Service ID:             S-1-5-21-1553278193-1769982261-2954363648-502

Network Information:
        Client Address:         ::ffff:10.0.0.6
        Client Port:            55912

Additional Information:
        Ticket Options:         0x40800010
        Result Code:            0x0
        Ticket Encryption Type: 0x12
        Pre-Authentication Type:        16

Certificate Information:
        Certificate Issuer Name:                aceresponder-CA-CA
        Certificate Serial Number:      690000004AF930C20ACBE28A8000000000004A
        Certificate Thumbprint:         EC64CC156B964A8C2A2575550F7492F4AA583E5D

Certificate information is only provided if a certificate was used for pre-authentication.

Pre-authentication types, ticket options, encryption types and result codes are defined in RFC 4120.
```

Forensics ဒါမှမဟုတ် incident investigation လုပ်နေတဲ့အချိန်မှာ “ဒီ certificate ကို attacker သုံးထားနိုင်တယ်” လို့ သံသယရှိလာရင် ဒီအချက်အလက်တွေကို အသုံးပြုပြီး စစ်ဆေးနိုင်ပါတယ်။ Domain Controller ပေါ်က 4768 event ထဲမှာပါတဲ့ certificate serial number နဲ့ thumbprint ကို ယူပြီး CA server ပေါ်မှာရှိတဲ့ Issued Certificates database နဲ့ cross-reference လုပ်နိုင်ပါတယ်။ CA database ထဲမှာလည်း certificate တစ်ခုချင်းစီရဲ့ serial number နဲ့ thumbprint ကို သိမ်းထားပါတယ်။

ဒီလိုနှိုင်းယှဉ်ကြည့်လိုက်ရင် အဲ့ဒီ certificate ကို ဘယ် user အတွက် ထုတ်ပေးထားတာလဲ၊ ဘယ် template ကို သုံးထားတာလဲ၊ ဘယ်အချိန်မှာ ထုတ်ပေးခဲ့တာလဲဆိုတာတွေကို ပြန်လည် သိနိုင်ပါတယ်။ အဲ့ဒါကြောင့် certificate-based authentication ဟာ attacker အတွက် အလွန်အင်အားကြီးတဲ့ နည်းလမ်းဖြစ်သလို defender အတွက်လည်း log correlation လုပ်ပြီး attacker ရဲ့ လှုပ်ရှားမှုကို trace လုပ်နိုင်တဲ့ အရေးကြီးတဲ့ အချက်အလက်တွေကို ပေးနိုင်ပါတယ်။

Test : ESC1 Auth

What is the certificate serial number for the ESC1 attack where garrett_hayes requested a certificate with sherrie_williams as the subject alternate name (SAN)?

ဒီမေးခွန်းမှာ စစ်ချင်တာက ESC1 attack တစ်ခုအတွင်း attacker က certificate ကို အသုံးပြုပြီး Kerberos authentication လုပ်ခဲ့တာကို သက်သေပြနိုင်မလားဆိုတာပါ။ အခြေအနေကို ပြန်လည် ချိတ်ဆက်ကြည့်ရင် garrett_hayes ဆိုတဲ့ user က certificate ကို request လုပ်ခဲ့ပြီး SAN ထဲမှာ sherrie_williams ကို ထည့်ထားပါတယ်။ ဒါဟာ ESC1 misconfiguration ကို abuse လုပ်ထားတဲ့ classic case တစ်ခုဖြစ်ပြီး attacker က ကိုယ်မဟုတ်တဲ့ user အဖြစ် authenticate လုပ်နိုင်သွားပါတယ်။

Certificate-based authentication ကို အသုံးပြုပြီး attacker က Kerberos Ticket Granting Ticket (TGT) ကို request လုပ်တဲ့အချိန် Domain Controller ပေါ်မှာ Event ID 4768 ဖြစ်တဲ့ security log ထွက်ပေါ်လာပါတယ်။ ဒီ event က Kerberos authentication စတင်တဲ့အခါ မှတ်တမ်းတင်တာဖြစ်ပြီး certificate ကို အသုံးပြုထားတဲ့အခါ certificate ရဲ့ serial number ကိုပါ log ထဲမှာ ထည့်ပေးပါတယ်။ ဒီ serial number က “ဘယ် certificate ကို သုံးထားလဲ” ဆိုတာကို ချိတ်ဆက်နိုင်တဲ့ အရေးကြီးဆုံး အချက်အလက် ဖြစ်ပါတယ်။

```
external_table('Winlog')
| where EventCode == 4768 and isnotempty(CertSerialNumber) and
TargetUserName contains "sherrie_williams"
| project Timestamp, Message
```

ပေးထားတဲ့ query ကို ကြည့်မယ်ဆိုရင် Winlog table ထဲက EventCode 4768 ကို ရွေးထားပြီး CertSerialNumber မဟုတ်တဲ့ field မဟုတ်ဘဲ certificate-based authentication ဖြစ်ကြောင်း အတည်ပြုဖို့ CertSerialNumber ပါတဲ့ event တွေကိုသာ စစ်ထားပါတယ်။ TargetUserName ကို sherrie_williams လို့ filter လုပ်ထားတာက attacker က sherrie_williams အဖြစ် authenticate ဝင်ထားတာကို ဖမ်းမိဖို့ပါ။ ဒီ query ရဲ့ output ထဲက Message field ကို ကြည့်ရင် certificate serial number ကို စာသားအနေနဲ့ တွေ့ရပါလိမ့်မယ်။ အဲ့ဒီ serial number ပဲ ဒီမေးခွန်းရဲ့ အဖြေ ဖြစ်ပါတယ်။

```
A Kerberos authentication ticket (TGT) was requested.

Account Information:
        Account Name:           SHERRIE_WILLIAMS
        Supplied Realm Name:    ACERESPONDER.LAB
        User ID:                        S-1-5-21-1553278193-1769982261-2954363648-3042

Service Information:
        Service Name:           krbtgt
        Service ID:             S-1-5-21-1553278193-1769982261-2954363648-502

Network Information:
        Client Address:         ::ffff:10.0.0.6
        Client Port:            51640

Additional Information:
        Ticket Options:         0x40800010
        Result Code:            0x0
        Ticket Encryption Type: 0x12
        Pre-Authentication Type:        16

Certificate Information:
        Certificate Issuer Name:                aceresponder-CA-CA
        Certificate Serial Number:      690000004F5FF0059A2537FDD500000000004F
        Certificate Thumbprint:         DA66D118968B0BC38CC45A8FDAC24277D1D604F5

Certificate information is only provided if a certificate was used for pre-authentication.

Pre-authentication types, ticket options, encryption types and result codes are defined in RFC 4120.
```

Lab environment မှာဆိုရင် image ထဲက log ကို ဖတ်ပြီး serial number ကို တိတိကျကျ ကူးထုတ်ရပါမယ်။ Real-world scenario မှာတော့ ဒီ serial number တစ်ခုတည်းနဲ့ ဆုံးဖြတ်တာ မလုံလောက်သေးပါဘူး။ Domain Controller log ထဲမှာ တွေ့တဲ့ serial number ကို CA server ပေါ်က Issued Certificates database နဲ့ cross-reference လုပ်ရပါမယ်။ CA ထဲမှာ အဲ့ဒီ serial number နဲ့ ကိုက်ညီတဲ့ certificate ကို ရှာပြီး Subject၊ SAN၊ template နဲ့ enrollment user ကို စစ်ကြည့်ရပါမယ်။ အဲ့ဒီအခါမှသာ “ဒီ certificate ဟာ garrett_hayes က request လုပ်ပြီး sherrie_williams ကို SAN အနေနဲ့ abuse လုပ်ထားတဲ့ ESC1 certificate အမှန်ပဲ” လို့ အတည်ပြုနိုင်ပါတယ်။

RPC Firewall extension ကို အသုံးပြုထားတဲ့ environment မှာဆိုရင် enrollment event နဲ့ TGT request event ကို အချိန်နီးကပ်မှု၊ source IP တူညီမှုနဲ့ ချိတ်ဆက်ပြီး အမြန်ဆုံး ခန့်မှန်းနိုင်ပါတယ်။ ဒါပေမယ့် forensic အမြင်နဲ့ဆိုရင် CA server ကို အမြဲ reference လုပ်တာက အကောင်းဆုံးနည်းလမ်း ဖြစ်ပါတယ်။ CA က “certificate အမှန်တကယ် ဘယ်လိုထုတ်ခဲ့လဲ” ဆိုတာကို authoritative အနေနဲ့ ပြောနိုင်တဲ့ တစ်ခုတည်းသော source ဖြစ်လို့ပါ။

ESC2 & ESC3

ESC2 နဲ့ ESC3 ဆိုတာတွေဟာ AD CS certificate template misconfiguration တွေကို abuse လုပ်ပြီး privilege escalation သို့မဟုတ် persistence ရယူနိုင်တဲ့ နည်းလမ်းတွေပါ။ ESC1 လို SAN ကို တိုက်ရိုက် abuse မလုပ်နိုင်တဲ့ template တွေထဲမှာတောင် အန္တရာယ်ရှိနိုင်တဲ့ EKU configuration တွေ ရှိနေတယ်ဆိုတာကို ဒီနေရာမှာ ပြထားပါတယ်။

ESC2

ESC2 က template က request လုပ်တဲ့သူကို Subject Alternative Name ကို ကိုယ်တိုင် သတ်မှတ်ခွင့်မပေးတဲ့အခြေအနေမှာတောင် အန္တရာယ်ရှိနိုင်တဲ့ EKU configuration တွေကို ဆိုလိုပါတယ်။ အထူးသဖြင့်

Any Purpose EKU နဲ့ EKU မပါတဲ့ certificate တွေဟာ လုံခြုံရေးအရ အလွန်အန္တရာယ်ကြီးပါတယ်။
Any Purpose EKU ပါတဲ့ certificate တစ်ခုဟာ “ဘာလုပ်မလဲဆိုတာ မကန့်သတ်ထားတဲ့ certificate” လို ပါပဲ။ ဒီလို certificate ကို အသုံးပြုတဲ့အခါ Client Authentication၊ Server Authentication စတဲ့ EKU တွေပါရှိသလို သုံးနိုင်သွားပါတယ်။ ဒါကြောင့် ဒီ certificate ကို တခြား technique တွေနဲ့ ပေါင်းလိုက်မယ်ဆိုရင် privilege escalation သို့မဟုတ် long-term persistence ရယူနိုင်ပါတယ်။ ထို့အပြင် Any Purpose EKU ပါတဲ့ certificate ဟာ Certificate Request Agent EKU ရှိသလိုပါပဲ အလုပ်လုပ်နိုင်တဲ့အတွက် ESC3 attack ကို ဆက်လက်လုပ်နိုင်တဲ့ အခြေခံအုတ်မြစ် ဖြစ်လာပါတယ်။

EKU မပါတဲ့ certificate ဆိုတာကလည်း အန္တရာယ်ရှိပါတယ်။ EKU မပါတဲ့ certificate တစ်ခုကို မည်သည့် ရည်ရွယ်ချက်အတွက်မဆို သုံးနိုင်ပြီး certificate အသစ်တွေကို sign လုပ်နိုင်ပါတယ်။ Default အခြေအနေမှာတော့ အဲ့ဒီ certificate နဲ့ sign လုပ်ထားတဲ့ certificate တွေကို authentication အတွက် သုံးခွင့်မရှိပါဘူး။ ဒါကြောင့် တိုက်ရိုက် privilege escalation ဖြစ်မလာပေမယ့် code signing စတဲ့ EKU တွေနဲ့ ပေါင်းလိုက်ရင် တခြား attack chain တွေအတွက် အသုံးချနိုင်ပါတယ်။

ESC3

ESC3 က ESC2 ထက် ပိုပြီး နက်ရှိုင်းပြီး နားလည်ရခက်တဲ့ concept တစ်ခုပါ။ ESC3 ရဲ့အဓိကအချက်က attacker က Certificate Request Agent EKU ပါတဲ့ certificate တစ်ခုကို ရရှိသွားတဲ့အခါ ဖြစ်ပါတယ်။ ဒီ EKU ပါတဲ့ certificate ကို ကိုင်ထားတဲ့သူဟာ “တခြား user အစား certificate လျှောက်ပေးနိုင်တဲ့ အခွင့်အရေး” ရရှိသွားပါတယ်။ ဆိုလိုတာက attacker က CA ကို “ဒီ administrator အတွက် certificate တစ်ခု ထုတ်ပေးပါ” လို့ တရားဝင်ပုံစံနဲ့ လျှောက်နိုင်သွားပါတယ်။

ဒီ feature က ဘာကြောင့်ရှိလဲဆိုတာကို နားလည်ရင် ESC3 ကို ပိုမိုနားလည်လွယ်ပါတယ်။ လက်တွေ့လုပ်ငန်းခွင်မှာ employee အသစ်တစ်ယောက် ဝင်လာတဲ့အချိန် administrator က smart card ကို ကိုယ်တိုင်ယူပြီး certificate enrollment လုပ်ပေးရတဲ့အခြေအနေတွေ ရှိပါတယ်။ User ကိုယ်တိုင် certificate လျှောက်မလုပ်နိုင်တဲ့အတွက် administrator က “ငါ ဒီ user အတွက် certificate လျှောက်ပေးနိုင်ပါတယ်” ဆိုတာကို CA ကို သက်သေပြဖို့ Certificate Request Agent EKU ပါတဲ့ certificate ကို အသုံးပြုပါတယ်။ အဲ့ဒီ certificate နဲ့ user ရဲ့ public key နဲ့ identity ကို sign လုပ်ပြီး CA ကို ပို့လိုက်တဲ့အခါ CA က “ဒီ request ကို လုပ်တဲ့သူဟာ တခြားသူအတွက် enrollment လုပ်ခွင့်ရှိတယ်” လို့ ယုံကြည်ပြီး certificate ကို ထုတ်ပေးပါတယ်။

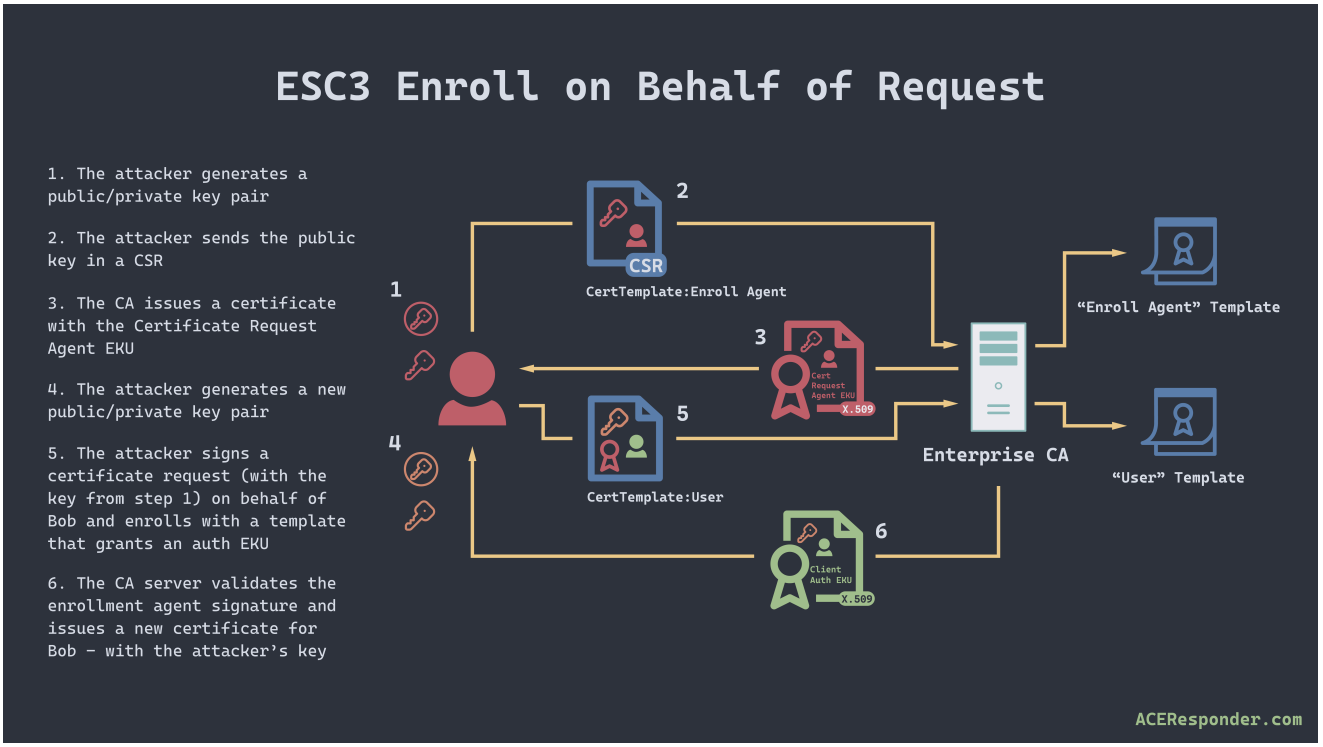
ESC3 attack မှာ certificate template နှစ်ခု ပါဝင်ပါတယ်။

- ပထမ template က attacker ကို Certificate Request Agent EKU ပါတဲ့ certificate တစ်ခု ရရှိစေပါတယ်။
- ဒုတိယ template က version 1 template ဖြစ်ရမယ် သို့မဟုတ် version 2 template ဖြစ်ပြီး authorized signatures issuance requirement ပါရမယ်။ ဒီ template က “တခြား certificate တစ်ခုက ဒီ request ကို sign လုပ်ထားရင် လက်ခံမယ်” ဆိုတဲ့ အဓိပ္ပါယ်ပါ။

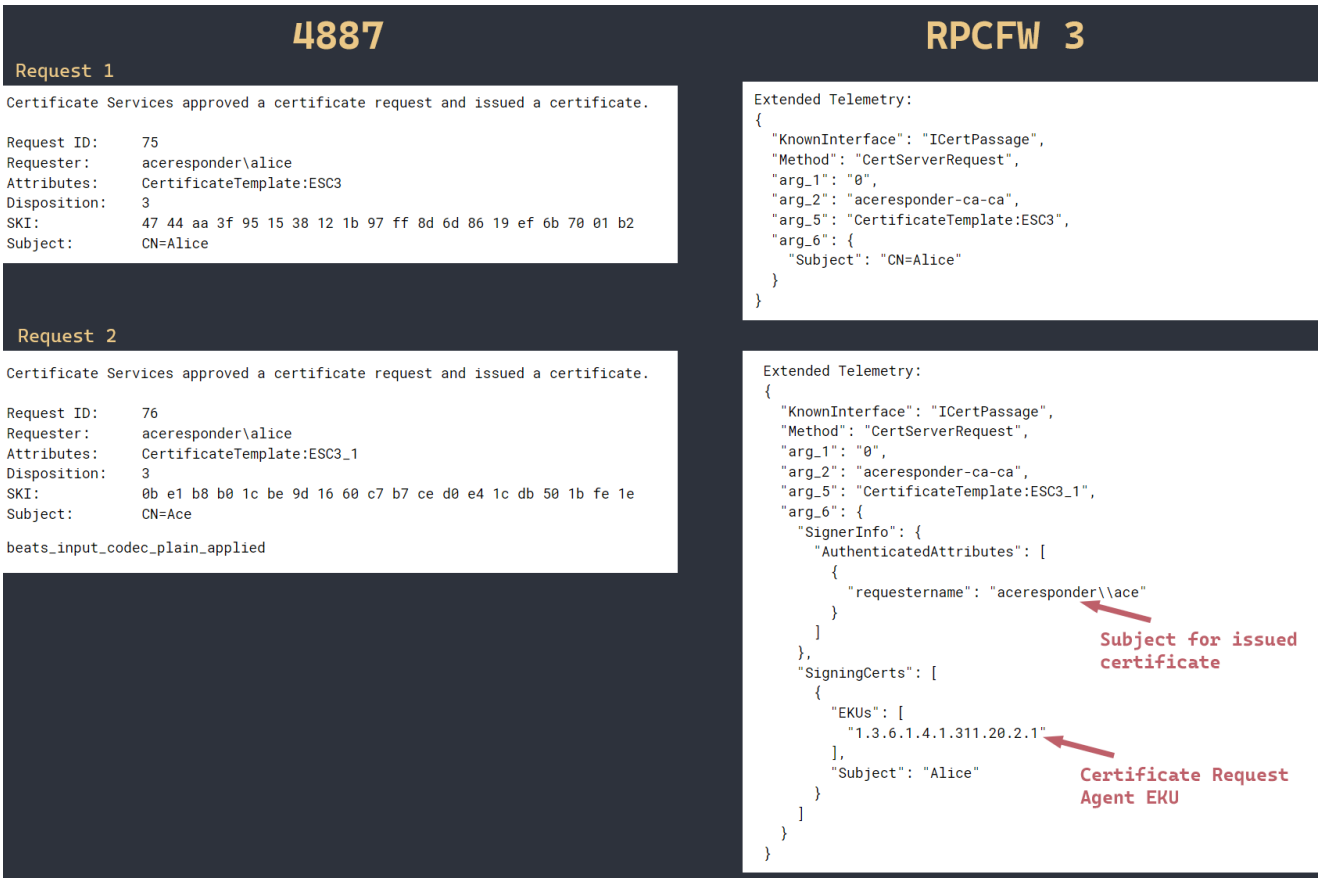
Attack sequence ကို ချိတ်ဆက်ပြီး စဉ်းစားကြည့်မယ်ဆိုရင် attacker က

1. ပထမဆုံး enrollment agent certificate ကို လျှောက်ပါတယ်။ အဲ့ဒီနောက်
2. attacker က administrator အတွက် certificate အသစ်တစ်ခုကို ကိုယ်တိုင် ဖန်တီးပြီး ပထမဆုံး ရထားတဲ့ certificate နဲ့ sign လုပ်ပါတယ်။ နောက်တစ်ဆင့်မှာ
3. attacker က အဲ့ဒီ administrator certificate ကို ဒုတိယ template ကို အသုံးပြုပြီး CA ဆီ လျှောက်ပါတယ်။ ဒီအချိန်မှာ ပထမဆုံး enrollment agent certificate ကို request ထဲမှာ ထည့်ပေးပါတယ်။

4. CA က request ကို စစ်တဲ့အခါ “ဒီ request ကို sign လုပ်ထားတဲ့ certificate မှာ Certificate Request Agent EKU ပါတယ်” လို့ တွေ့ရင် administrator certificate ကို တရားဝင်ထုတ်ပေးလိုက်ပါတယ်။
5. အဲ့ဒီနောက် attacker က administrator certificate ကို အသုံးပြုပြီး authentication လုပ်နိုင်သွားပါတယ်။



Log တွေထဲမှာ ဒီလို ESC3 attack ကို extended telemetry နဲ့ကြည့်ရင် request ကို sign လုပ်ထားတဲ့ certificate အကြောင်းကိုပါ တွေ့နိုင်ပါတယ်။ Signing cert ရဲ့ EKU ကို decode လုပ်ထားတဲ့အတွက် ဒီ request ဟာ Request On Behalf Of (ROBO) ဖြစ်ကြောင်းကို အတည်ပြုနိုင်ပါတယ်။ requestername field ထဲမှာ CA က certificate ရဲ့ Subject အဖြစ် သုံးရမယ့် user name ကို သိနိုင်ပါတယ်။



နောက်ဆုံးအနေနဲ့ သတိထားရမယ့် အချက်တစ်ခုက attacker က enrollment agent certificate ကို ကိုယ်တိုင် လျှောက်ယူရမယ်လို့ မဆိုလိုပါဘူး။ Enrollment agent certificate တစ်ခုကို compromise လုပ်ထားနိုင်လည်း ESC3 attack ကို လုပ်နိုင်ပါတယ်။ SpecterOps ရဲ့ research အရ computer account တချို့ကို enrollment agent template တွေမှာ enrollment rights ပေးထားတာ အလွန်အများအပြား တွေ့ရပြီး အဲ့ဒီ computer account compromise ဖြစ်သွားရင် ESC3 attack ကို အလွယ်တကူ ဆက်လုပ်နိုင်ပါတယ်။

ဒီအပိုင်းကို နားလည်ထားခြင်းက AD CS abuse ကို အဆင့်မြင့်အနေနဲ့ နားလည်ဖို့ အရေးကြီးပြီး ESC1 ကနေ ESC3 အထိ ဘယ်လို attack chain တွေ ချိတ်ဆက်နိုင်လဲဆိုတာကို မြင်နိုင်အောင် ကူညီပေးပါတယ်။

Test : ESC3

An attacker used a vulnerable certificate template (not ESC3) to get a certificate with a Certificate Request Agent EKU. Which template did the attacker abuse?

ဒီ ESC3 scenario မှာ attacker က **ESC3 template ကို တိုက်ရိုက် abuse မလုပ်ပါဘူး**။ အစား **အခြား vulnerable template တစ်ခုကို abuse လုပ်ပြီး Certificate Request Agent EKU ပါတဲ့ certificate တစ်ခုကို ရယူခဲ့တာ** ဖြစ်ပါတယ်။ အဲ့ဒီ certificate ကို အသုံးပြုပြီး ROBO (Request On Behalf Of) attack ကို ဆက်လုပ်သွားပါတယ်။

အရင်ဆုံး ESC3 detection logic ကို နားလည်ဖို့လိုပါတယ်။ ESC3 attack မှာ အရေးကြီးဆုံး indicator က **certificate request တစ်ခုထဲမှာ signing certificate ပါလာခြင်း** ဖြစ်ပါတယ်။ Signing certificate ပါလာတယ်ဆိုတာက “ဒီ certificate request ကို တခြား certificate တစ်ခုက sign လုပ်ထား

တယ်” ဆိုတဲ့ အဓိပ္ပါယ်ပါ။ ဒါက Certificate Request Agent EKU ကို အသုံးပြုထားတဲ့ ROBO request ဖြစ်နိုင်တယ်ဆိုတဲ့ အချက်ကို ပြပါတယ်။

RPCFW telemetry ကို အသုံးပြုတဲ့အခါ SigningCerts field ပါတဲ့ EventCode 3 ကိုရှာလိုက်ရင် ROBO request တွေကို တွေ့နိုင်ပါတယ်။

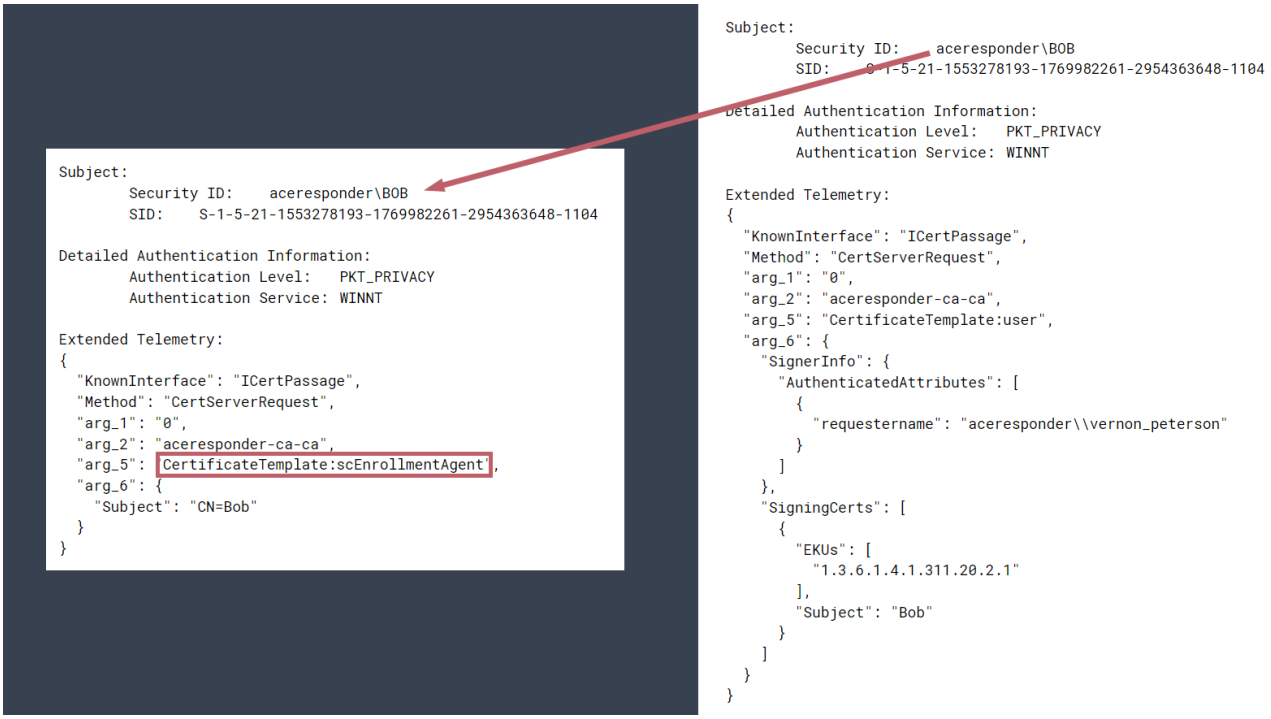
```
external_table('Winlog')
| where WinlogProviderName == "RPCFW" and EventCode == 3 and
isnotempty(Arg6.SigningCerts)
| project Timestamp, Message
```

ဒီနေရာမှာ Bob က request လုပ်ထားတဲ့ event ကို တွေ့ရပါတယ်။ အရေးကြီးတာက Bob ဟာ ပုံမှန် အားဖြင့် “တခြားသူအတွက် certificate လျှောက်ပေးနိုင်တဲ့ administrator မဟုတ်ဘူး” ဆိုတာပါ။Bob က administrator မဟုတ်ဘဲ **User template** ကို အသုံးပြုပြီး Certificate Request Agent EKU ပါတဲ့ certificate ကို ရယူနိုင်ခဲ့တယ်။Normally User template က ROBO request အတွက် သုံးမယ့် template မဟုတ်ပါဘူး။ ဒါပေမယ့် misconfiguration ကြောင့် User template ထဲမှာ Any Purpose EKU သို့မဟုတ် Certificate Request Agent EKU ပါနေတဲ့အတွက် attacker က abuse လုပ်နိုင်ခဲ့တာ ဖြစ်ပါတယ်။

နောက်တစ်ဆင့်မှာ Bob ကိုယ်တိုင် certificate request လုပ်ထားတဲ့ event ကို ဆက်ရှာကြည့်ပါတယ်။

```
external_table('Winlog')
| where WinlogProviderName == "RPCFW" and EventCode == 3 and Arg6 contains
"bob"
| project Timestamp, Message
```

RPCFW EventCode 3 ထဲမှာ Arg6 မှာ “bob” ပါတဲ့ request ကို တွေ့ရပါတယ်။ ဒီ events နှစ်ခုက အချိန်အလွန်နီးကပ်ပြီး ဖြစ်ပေါ်လာတာကြောင့် attack chain တစ်ခုထဲက sequence ဖြစ်တယ်လို့ ယူဆနိုင်ပါတယ်။



Investigator က AD CS 4887 events ကို သေချာဖတ်ပြီး “ဘယ်သူလျှောက်လဲ” နဲ့ “certificate က ဘယ်သူအတွက်လဲ” ကို မတူညီဘူးဆိုရင် အဲ့ဒီ request ကို ROBO attack အဖြစ် သံသယထားပြီး ဆက်လက် စုံစမ်းရမယ်။

ESC4 - Template Modification

ESC4 ဆိုတာက **certificate template ကို တိုက်ရိုက် ပြင်ဆင်ပြီး attack လုပ်တဲ့နည်းလမ်း** ဖြစ်ပါတယ်။ AD CS ထဲမှာ certificate templates တွေဟာ **Active Directory object တွေလိုပဲ security permissions ရှိတဲ့ object တွေ** ဖြစ်ပါတယ်။ အဲ့ဒီကြောင့် attacker တစ်ယောက်က template ကို control လုပ်နိုင်တဲ့ permission ရှိရင် user account, group, GPO စတဲ့ AD object တွေကို abuse လုပ်သလိုပဲ template ကိုပါ abuse လုပ်နိုင်ပါတယ်။

SpecterOps ရဲ့ white paper မှာ ဖော်ပြထားတဲ့ ESC4 attack က **template တစ်ခုကို ကိုယ်တိုင် vulnerable ဖြစ်အောင် ပြင်လိုက်ပြီး ESC1 attack ကို ဖန်တီးလိုက်တာ** ဖြစ်ပါတယ်။ ပုံမှန်အားဖြင့် template တစ်ခုက SAN ကို user ကိုယ်တိုင် supply လုပ်ခွင့် မပေးထားရင် ESC1 မဖြစ်နိုင်ပါဘူး။ ဒါပေမယ့် attacker က template setting ကို ပြင်လိုက်ရင် ESC1 ဖြစ်လာနိုင်ပါတယ်။

ESC1 behavior ကို activate လုပ်ပေးတဲ့ အဓိက setting က template ထဲက **CT_FLAG_ENROLLEE_SUPPLIES_SUBJECT** ဖြစ်ပါတယ်။ ဒီ flag ကို **ms-PKI-Certificate-Name-Flag** ဆိုတဲ့ attribute ထဲမှာ သိမ်းထားပါတယ်။ ဒီ flag ကို enable လုပ်လိုက်တာနဲ့ certificate လျှောက်တဲ့ user က Subject သို့မဟုတ် SAN ကို ကိုယ်တိုင် ထည့်ခွင့်ရသွားပါတယ်။ ဒါက privilege escalation အတွက် အလွန်အန္တရာယ်ကြီးတဲ့ configuration ဖြစ်ပါတယ်။

Template object ကို abuse လုပ်နိုင်တဲ့ နည်းလမ်းတွေက တခြားလည်း ရှိပေမယ့် **template ကို ESC1 vulnerable ဖြစ်အောင် ပြင်ပြီး SAN attack လုပ်တာက အများဆုံးတွေ့ရတဲ့ နည်းလမ်း** ဖြစ်ပါတယ်။ အကြောင်းက PKI administration က အရမ်းရှုပ်ထွေးပြီး permission တွေကို မှားယွင်းပေးထားတာ မကြာခဏ ဖြစ်လေ့ရှိလို့ပါ။

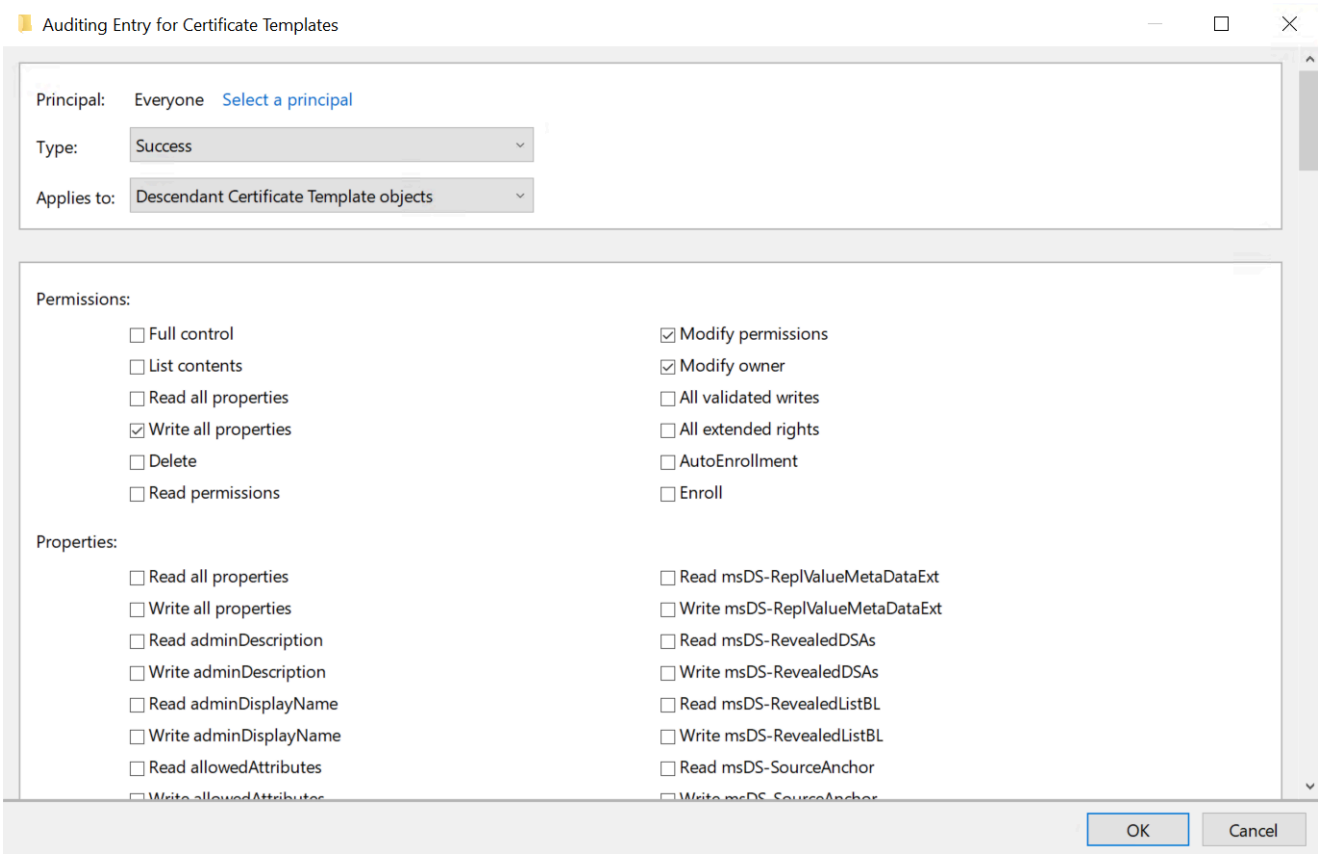
Windows အမြင်အရ template ကို ပြင်ဆင်လိုက်ရင် Security log ထဲမှာ event တွေ ထွက်သင့်ပါတယ်။ Template property ကို ပြင်လိုက်ရင် Event ID 4899 ထွက်ရပြီး template ရဲ့ security descriptor ကို ပြင်ရင် Event ID 4900 ထွက်ရပါမယ်။ ဒါပေမယ့် လက်တွေ့မှာတော့ ဒီ events တွေဟာ **မကြာခဏ မထွက်ဘဲ လွတ်သွားတာ** တွေ့ရပါတယ်။ ဒါကြောင့် ESC4 detection က မလွယ်ပါဘူး။

ဒီပြဿနာကို ဖြေရှင်းဖို့ Certificate Templates container ပေါ်မှာ **SACL (System Access Control List)** ကို သတ်မှတ်ထားပါတယ်။

> SACL ဆိုတာက “ဒီ object ကို ဘယ်သူ ဘာလုပ်လဲဆိုတာကို audit log ထဲမှာ မှတ်တမ်းတင်ပေး” ဆိုတဲ့ setting ဖြစ်ပါတယ်။

ဒီ audit ကို အလုပ်လုပ်စေဖို့ Windows မှာ **Audit Directory Service Changes** ဆိုတဲ့ policy ကို enable လုပ်ပြီး Event ID 4662 ကို ဖမ်းရပါတယ်။

SACL ကို `Configuration > Services > Public Key Services > Certificate Templates container` ပေါ်မှာ သတ်မှတ်လိုက်ရင် template object တစ်ခုခု ပြင်ဆင်တိုင်း audit log ထဲမှာ မှတ်တမ်းတင်ပေးပါတယ်။ ဒီလိုနဲ့ attacker က template ကို ပြင်ဆင်လိုက်တာကို catch လုပ်နိုင်ပါတယ်။



ESC4 attack ကို detect လုပ်တဲ့အခါ အထူးစောင့်ကြည့်ရမယ့် attribute က **ms-PKI-Certificate-Name-Flag** ဖြစ်ပါတယ်။ ဒီ attribute ရဲ့ GUID က **ea1dddc4-60ff-416e-8cc0-17cee534bce7** ဖြစ်ပြီး SAN supply ကို control လုပ်တဲ့ setting ပါဝင်ပါတယ်။ ဒီ attribute ပြောင်းလဲသွားတာကို တွေ့ရင် template ကို ESC1 vulnerable ဖြစ်အောင် ပြင်လိုက်တာ ဖြစ်နိုင်ပါတယ်။

```
An operation was performed on an object.

Subject :
        Security ID:            S-1-5-21-1553278193-1769982261-2954363648-1103
        Account Name:           alice
        Account Domain:         aceresponder
        Logon ID:               0x5FB5EB2

Object:
        Object Server:          DS
        Object Type:            %{e5209ca2-3bba-11d2-90cc-00c04fd91ab1}
        Object Name:            %{5d95058c-0991-4345-bcf2-4f146e59625e}
        Handle ID:              0x0

Operation:
        Operation Type:         Object Access
        Accesses:               Write Property

        Access Mask:            0x20
        Properties:             Write Property
        {e5209ca2-3bba-11d2-90cc-00c04fd91ab1}
                {771727b1-31b8-4cdf-ae62-4fe39fadf89e}
                        {bf967976-0de6-11d0-a285-00aa003049e2}
                        {f0bfdefa-3b9d-11d2-90cc-00c04fd91ab1}
                        {fc5a9106-3b9d-11d2-90cc-00c04fd91ab1}
                        {18976af6-3b9e-11d2-90cc-00c04fd91ab1}
                        {d15ef7d8-f226-46db-ae79-b34e560bd12c}
                        {ea1dddc4-60ff-416e-8cc0-17cee534bce7}
                        {dbd90548-aa37-4202-9966-8c537ba5ce32}


Additional Information:
        Parameter 1:            -
        Parameter 2:
```

Log ထဲမှာ တွေ့ရတဲ့ အခြေအနေက Alice ဆိုတဲ့ user က administrator မဟုတ်ပဲ template ကို overwrite လုပ်ထားတာပါ။ Windows က object name ကို မမှတ်တမ်းတင်ဘဲ **object GUID ပဲ မှတ်တမ်းတင်ထားတာ** ကြောင့် “ဘယ် template ကို ပြင်လဲ” ဆိုတာကို တိုက်ရိုက် မမြင်ရပါဘူး။ ဒါပေမယ့် ms-PKI-Certificate-Name-Flag ပြောင်းသွားတဲ့အချက်ကို ကြည့်ပြီး “ဒါ certificate template တစ်ခု ဖြစ်တယ်” ဆိုတာကို ခန့်မှန်းနိုင်ပါတယ်။

ဒီ template modification ပြီးသွားတဲ့နောက်မှာ certificate request တစ်ခု ထပ်ထွက်လာပြီး အဲဒီ request က **ESC1 attack နဲ့ လုံးဝဆင်တူတဲ့ behavior** ကို ပြသပါတယ်။ အဓိပ္ပါယ်က attacker က template ကို vulnerable ဖြစ်အောင် ပြင်ပြီးပြီးချင်း SAN ကို abuse လုပ်ပြီး privilege escalation ကို ဆက်လုပ်သွားတာ ဖြစ်ပါတယ်။

ESC4

Which user (other than alice) exploited ESC4?

ESC4 investigation မှာ အရင်ဆုံး စစ်ဆေးရမယ့်အချက်က **certificate template ကို ဘယ်သူတွေ ပြင်ခဲ့လဲ** ဆိုတာပါ။ ESC4 ရဲ့အဓိက သဘောတရားက template ကို vulnerable ဖြစ်အောင် ပြင်လိုက်ခြင်း ဖြစ်လို့၊ template modification ကို ဖမ်းနိုင်ရင် attack chain ကို စတင်တွေ့နိုင်ပါတယ်။

```
external_table('Winlog')
| where EventCode == 4662 and Properties contains "ea1dddc4-60ff-416e-8cc0-17cee534bce7"
| project Timestamp, SubjectUserName, Message
```

ဒီအတွက် Event ID 4662 ကို အသုံးပြုပါတယ်။ ဒီ event က Active Directory object တစ်ခုရဲ့ attribute ကို ပြောင်းလဲလိုက်တဲ့အခါ audit log ထဲမှာ ထွက်လာတဲ့ event ဖြစ်ပါတယ်။ Query ထဲမှာ **ms-PKI-Certificate-Name-Flag** ရဲ့ GUID ဖြစ်တဲ့ `ea1dddc4-60ff-416e-8cc0-17cee534bce7` ကို filter လုပ်ထားတာက ESC1 behavior ကို enable လုပ်ပေးတဲ့ attribute ကို တိတိကျကျ စောင့်ကြည့်ဖို့ပါ။

ဒီ query ရဲ့ရလဒ်အရ **Alice, Carol နဲ့ Ace** ဆိုတဲ့ user သုံးယောက်က ဒီ attribute ကို ပြင်ဆင်ထားတာကို တွေ့ရပါတယ်။ ဒါက “ဒီသုံးယောက်ထဲက တစ်ယောက်မဟုတ် တစ်ယောက်က ESC4 attack ကို ဆက်လုပ်သွားနိုင်တယ်” ဆိုတဲ့ အချက်ကို ပြပါတယ်။

Ace က domain administrator ဖြစ်တဲ့အတွက် certificate template ကို ပြင်ဆင်နိုင်တာဟာ ပုံမှန် administration activity ဖြစ်နိုင်ပါတယ်။ ဒါကြောင့် Ace ကို attack suspect အဖြစ် မယူဆပါဘူး။ Alice ကလည်း အရင်အပိုင်းတွေမှာ template modification လုပ်ထားတာကို တွေ့ထားပြီးသား ဖြစ်ပါတယ်။

အဲ့ဒီနောက်မှာ investigator က **Carol ကို အထူးသံသယထားပြီး** သူမရဲ့ certificate requests တွေကို ဆက်စစ်ပါတယ်။ ဒါကြောင့် RPCFW telemetry ထဲက EventCode 3 ကို အသုံးပြုပြီး Arg6 ထဲမှာ “Carol” ပါတဲ့ certificate request တွေကို ရှာပါတယ်။

```
external_table('Winlog')
| where WinlogProviderName == "RPCFW" and EventCode == 3 and Arg6 contains
"Carol"
| project Timestamp, Message
```

ဒီ query ရဲ့ရလဒ်အရ Carol က **template modification ပြီးပြီးချင်း SAN ပါတဲ့ certificate ကို လျှောက်ထားတာ** ကို တွေ့ရပါတယ်။ ဒီ certificate request က behavior အရ **ESC1 enrollment နဲ့ လုံးဝကိုက်ညီ** ပါတယ်။ အဓိပ္ပါယ်က Carol က template ကို vulnerable ဖြစ်အောင် ပြင်ထားတဲ့ အခြေအနေကို ချက်ချင်း အသုံးချပြီး privilege escalation attack ကို လုပ်သွားတာ ဖြစ်ပါတယ်။

Time	winlog.event_data.SubjectUserName
May 8, 2024 @ 21:10:05.351	alice
May 8, 2024 @ 21:08:53.367	carol
May 8, 2024 @ 21:07:26.103	ace
May 8, 2024 @ 20:22:29.052	alice
May 8, 2024 @ 20:22:15.812	alice

```
Subject:
        Security ID:    aceresponder\CAROL
        SID:    S-1-5-21-1553278193-1769982261-2954363648-1105

Detailed Authentication Information:
        Authentication Level:   PKT_PRIVACY
        Authentication Service: WINNT

Extended Telemetry:
{
  "KnownInterface": "ICertPassage",
  "Method": "CertServerRequest",
  "arg_1": "0",
  "arg_2": "aceresponder-CA-CA",
  "arg_5": "CertificateTemplate:userauth\nSAN:upn=ace@aceresponder.lab",
  "arg_6": {
    "Subject": "CN=Carol",
    "SubjectAltNames": [
      "ace@aceresponder.lab"
    ]
  }
}
```

ဒါကြောင့် ဒီ ESC4 scenario ကို အပြည့်အစုံ ချိတ်ဆက်ကြည့်မယ်ဆိုရင်
Template modification ကို Alice, Carol, Ace တို့ လုပ်ထားပေမယ့်

Domain admin ဖြစ်တဲ့ Ace ကို ဖယ်ရှားပြီး
ESC1-style certificate request ကို အမှန်တကယ် လုပ်သွားတဲ့ user က **Carol** ဖြစ်ပါတယ်။
