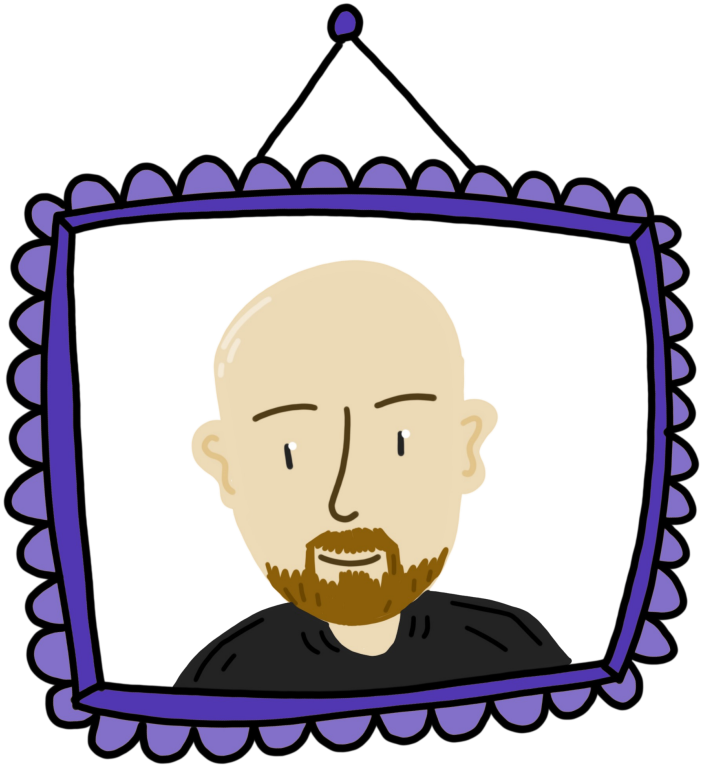
I've Got a Golden Twinkle In My Eye

SANS Pentest HackFest 2022

Charlie Clark & Andrew Schwartz

TODAY'S TOPICS

- INTRODUCTIONS
- KERBEROS **101**
- GOLDEN TICKET **101**
- DETECTION METHODS
- DEMOS



CHARLIE CLARK

- @EXPLOITPH
- SECURITY RESEARCHER @ SEMPERIS
- ATTACKER OF KERBEROS
- STREAMING MEDIA AFICIONADO

ANDREW SCHWARTZ

- @4NDR3W6S
- PRACTICE LEAD @ TRUSTEDSEC
- ATTACK/DETECT ALL THE THINGS
- TOTTENHAM SUPER FAN (COYS!)

A FEW DISCLAIMERS

- ASSUMPTIONS, SCOPE, AND BLIND SPOTS
- AD IS NOT "STATIC"
- WE WILL BE EXPLICIT ON WHAT WE CAN DETECT
- ANY DETECTION CAN BY BYPASSED
- WE DO NOT "STOP" GOLDEN TICKETS
- WE ARE NOT FOCUSING ON SILVER TICKETS

Welcome to Kerberos 101!

1 USERNAME: CBUCKET
PASSWORD: GOLD3NTICK3T

LOGIN

5

LSA/LSASS

2

3

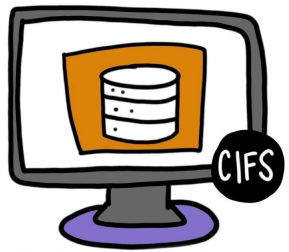
6

4

7

8

dc.chocolatefactory.local
(DOMAIN CONTROLLER)



fs.chocolatefactory.local
(SERVER)

ORIGINAL CONCEPT & INSPIRATION FROM BENJAMIN DELPY & WILL SCHROEDER

1 USERNAME: CBUCKET
PASSWORD: GOLD3NTICK3T

LOGIN

5

LSA/LSASS

2 CREATE HASH FROM CREDENTIALS:
RC4 - USER'S PASSWORD
AES - PASSWORD + SALT

3

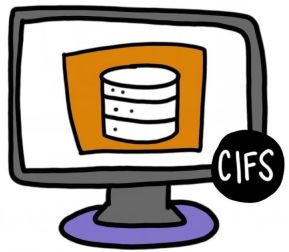
6

4

7

8

dc.chocolatefactory.local
(DOMAIN CONTROLLER)



fs.chocolatefactory.local
(SERVER)

ORIGINAL CONCEPT & INSPIRATION FROM BENJAMIN DELPY & WILL SCHROEDER

1 USERNAME: CBUCKET
PASSWORD: GOLD3NTICK3T

LOGIN

5

LSA/LSASS

2 CREATE HASH FROM CREDENTIALS:
RC4 - USER'S PASSWORD
AES - PASSWORD + SALT

3 ENCRYPT PART OF REQUEST WITH HASH

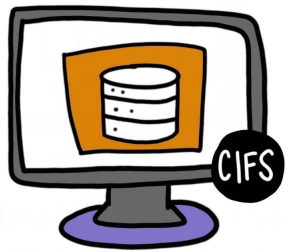
6

4

7

8

dc.chocolatefactory.local
(DOMAIN CONTROLLER)



fs.chocolatefactory.local
(SERVER)

ORIGINAL CONCEPT & INSPIRATION FROM BENJAMIN DELPY & WILL SCHROEDER

1 USERNAME: CBUCKET
PASSWORD: GOLD3NTICK3T

LOGIN

5

LSA/LSASS

2 CREATE HASH FROM CREDENTIALS:
RC4 - USER'S PASSWORD
AES - PASSWORD + SALT

3 ENCRYPT PART OF REQUEST WITH HASH

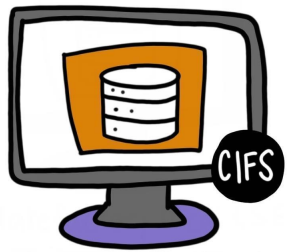
6

4 TGT RETRIEVAL

7

8

dc.chocolatefactory.local
(DOMAIN CONTROLLER)



fs.chocolatefactory.local
(SERVER)

ORIGINAL CONCEPT & INSPIRATION FROM BENJAMIN DELPY & WILL SCHROEDER

1 USERNAME: CBUCKET
PASSWORD: GOLD3NTICK3T

LOGIN

5 ACCESS ATTEMPT/ACTION

dir \\fs.chocolatefactory.local\C$

LSA/LSASS

2 CREATE HASH FROM CREDENTIALS:
RC4 - USER'S PASSWORD
AES - PASSWORD + SALT

3 ENCRYPT PART OF REQUEST WITH HASH

6

4 TGT RETRIEVAL

7

8

dc.chocolatefactory.local
(DOMAIN CONTROLLER)

fs.chocolatefactory.local
(SERVER)

ORIGINAL CONCEPT & INSPIRATION FROM BENJAMIN DELPY & WILL SCHROEDER

1 USERNAME: CBUCKET
PASSWORD: GOLD3NTICK3T

LOGIN

5 ACCESS ATTEMPT/ACTION

dir\\fs.chocolatefactory.local\c$

LSA/LSASS

2 CREATE HASH FROM CREDENTIALS:
RC4 - USER'S PASSWORD
AES - PASSWORD + SALT

3 ENCRYPT PART OF REQUEST WITH HASH

6 USE TGT TO REQUEST SERVICE TICKET

4 TGT RETRIEVAL

7

8

dc.chocolatefactory.local
(DOMAIN CONTROLLER)

CIFS

fs.chocolatefactory.local
(SERVER)

ORIGINAL CONCEPT & INSPIRATION FROM BENJAMIN DELPY & WILL SCHROEDER

1 USERNAME: CBUCKET
PASSWORD: GOLD3NTICK3T

LOGIN

5 ACCESS ATTEMPT/ACTION

dir \\fs.chocolatefactory.local\C$

LSA/LSASS

2 CREATE HASH FROM CREDENTIALS:
RC4 – USER'S PASSWORD
AES- PASSWORD + SALT

3 ENCRYPT PART OF REQUEST WITH HASH

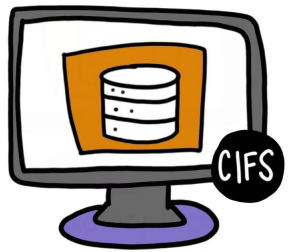
6 USE TGT TO REQUEST SERVICE TICKET

4 TGT RETRIEVAL

7 SERVICE TICKET RETRIEVAL

8

dc.chocolatefactory.local
(DOMAIN CONTROLLER)



fs.chocolatefactory.local
(SERVER)

ORIGINAL CONCEPT & INSPIRATION FROM BENJAMIN DELPY & WILL SCHROEDER

1 USERNAME: CBUCKET
PASSWORD: GOLD3NTICK3T

LOGIN

5 ACCESS ATTEMPT/ACTION

dir \\fs.chocolatefactory.local\C$

LSA/LSASS

2 CREATE HASH FROM CREDENTIALS:
RC4 - USER'S PASSWORD
AES - PASSWORD + SALT

3 ENCRYPT PART OF REQUEST WITH HASH

6 USE TGT TO REQUEST SERVICE TICKET

4 TGT RETRIEVAL

7 SERVICE TICKET RETRIEVAL

8 SERVICE ACCESS

dc.chocolatefactory.local
(DOMAIN CONTROLLER)

fs.chocolatefactory.local
(SERVER)

ORIGINAL CONCEPT & INSPIRATION FROM BENJAMIN DELPY & WILL SCHROEDER

1 USERNAME: CBUCKET
PASSWORD: GOLD3NTICK3T

LOGIN

5 ACCESS ATTEMPT/ACTION

dir \\fs.chocolatefactory.local\C$

LSA/LSASS

2 CREATE HASH FROM CREDENTIALS:
RC4 - USER'S PASSWORD
AES - PASSWORD + SALT

3 ENCRYPT PART OF REQUEST WITH HASH

6 USE TGT TO REQUEST SERVICE TICKET

4 TGT RETRIEVAL
(AS-REQ/AS-REP)

7 SERVICE TICKET RETRIEVAL
(TGS-REQ/TGS-REP)

8 SERVICE ACCESS
(AP-REQ/AP-REP)

dc.chocolatefactory.local
(DOMAIN CONTROLLER)

fs.chocolatefactory.local
(SERVER)

ORIGINAL CONCEPT & INSPIRATION FROM BENJAMIN DELPY & WILL SCHROEDER

Lets talk Golden Tickets!

Lets talk Golden Tickets!

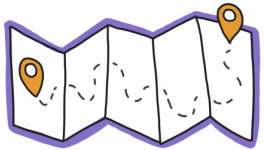
What is a Golden Ticket?

A Golden Ticket is any TGT that is NOT ISSUED by a KDC

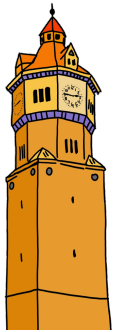
Attacker Motives:

GOLDEN TICKET 101

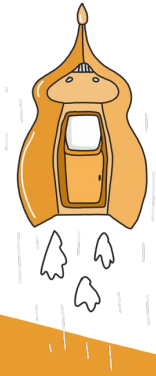
Impersonation



Unfettered Access

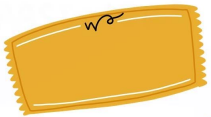


Persistence



Privilege escalation within a Forest

1 CREATION OF GOLDEN TICKET (GT)



2

3

LSA/LSASS

4

5

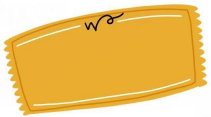
6

dc.chocolatefactory.local
(DOMAIN CONTROLLER)

fs.chocolatefactory.local
(SERVER)

ORIGINAL CONCEPT & INSPIRATION FROM BENJAMIN DELPY & WILL SCHROEDER

1 CREATION OF GOLDEN TICKET (GT)



2 INJECTION OF GT

LSA/LSASS

4

3

5

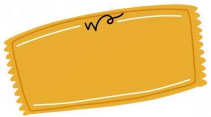
6

dc.chocolatefactory.local
(DOMAIN CONTROLLER)

fs.chocolatefactory.local
(SERVER)

ORIGINAL CONCEPT & INSPIRATION FROM BENJAMIN DELPY & WILL SCHROEDER

1 CREATION OF GOLDEN TICKET (GT)



2 INJECTION OF GT

LSA/LSASS

4

3 ACCESS ATTEMPT/ACTION

dir\\fs.chocolatefactory.local\c$

6

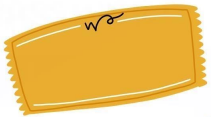
5

dc.chocolatefactory.local
(DOMAIN CONTROLLER)

fs.chocolatefactory.local
(SERVER)

ORIGINAL CONCEPT & INSPIRATION FROM BENJAMIN DELPY & WILL SCHROEDER

1 CREATION OF GOLDEN TICKET (GT)



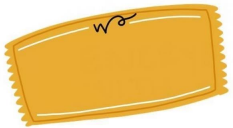
2 INJECTION OF GT

3 ACCESS ATTEMPT/ACTION

dir\\fs.chocolatefactory.local\c$

LSA/LSASS

4 USE GOLDEN TICKET TO



REQUEST SERVICE TICKET

5

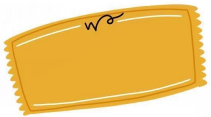
6

dc.chocolatefactory.local
(DOMAIN CONTROLLER)

fs.chocolatefactory.local
(SERVER)

ORIGINAL CONCEPT & INSPIRATION FROM BENJAMIN DELPY & WILL SCHROEDER

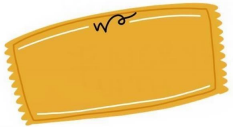
1 CREATION OF GOLDEN TICKET (GT)



2 INJECTION OF GT

LSA/LSASS

4 USE GOLDEN TICKET TO



REQUEST SERVICE TICKET

3 ACCESS ATTEMPT/ACTION

dir\\fs.chocolatefactory.local\c$

6

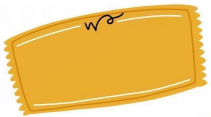
5 SERVICE TICKET RETRIEVAL (TGS-REQ/TGS-REP)

dc.chocolatefactory.local
(DOMAIN CONTROLLER)

fs.chocolatefactory.local
(SERVER)

ORIGINAL CONCEPT & INSPIRATION FROM BENJAMIN DELPY & WILL SCHROEDER

1 CREATION OF GOLDEN TICKET (GT)



2 INJECTION OF GT

LSA/LSASS

4 USE GOLDEN TICKET TO

REQUEST SERVICE TICKET

3 ACCESS ATTEMPT/ACTION

dir\\fs.chocolatefactory.local\c$

5 SERVICE TICKET RETRIEVAL
(TGS-REQ/TGS-REP)

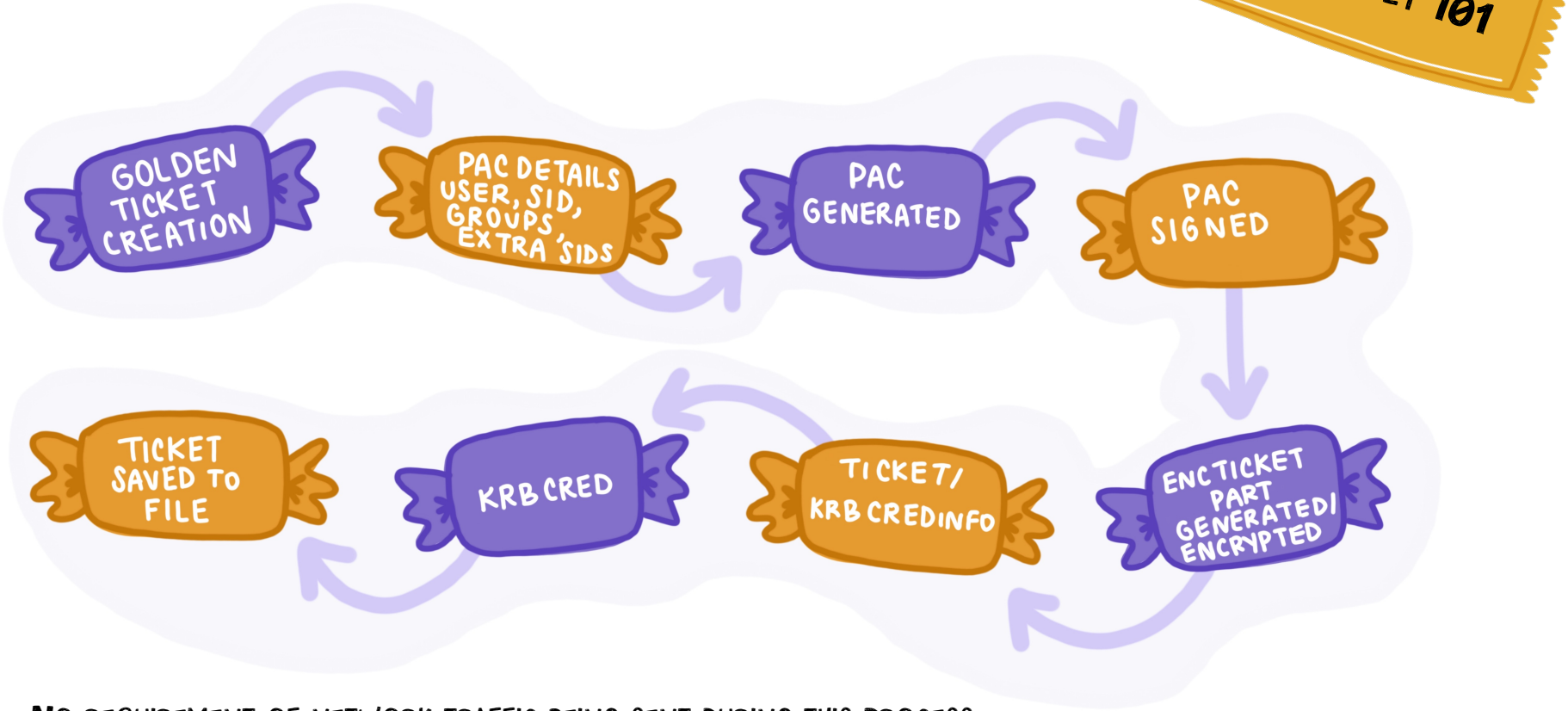
6 SERVICE ACCESS
(AP-REQ/AP-REP)

dc.chocolatefactory.local
(DOMAIN CONTROLLER)

fs.chocolatefactory.local
(SERVER)

ORIGINAL CONCEPT & INSPIRATION FROM BENJAMIN DELPY & WILL SCHROEDER

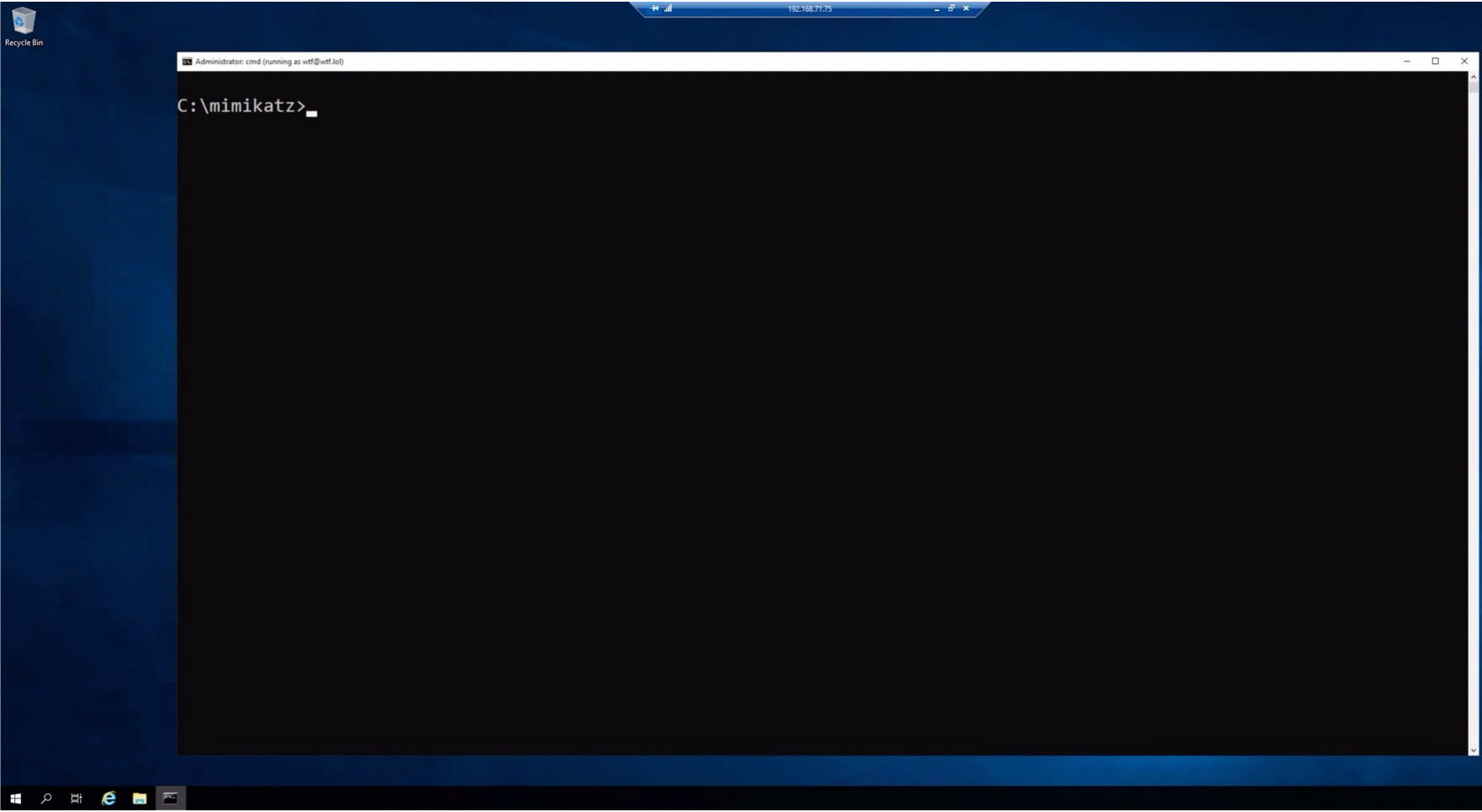
Golden Ticket Creation Flow



No requirement of network traffic being sent during this process

DEMO

DEMO 1: GOLDEN TICKET CREATION & USAGE


192.168.71.75
Administrator: cmd (running as wtf@wtf.lol)
C:\mimikatz>
Recycle Bin
23:01
05/11/2022

Current Detection Methods & Drawbacks

EVTX 4768 EVTX 4769

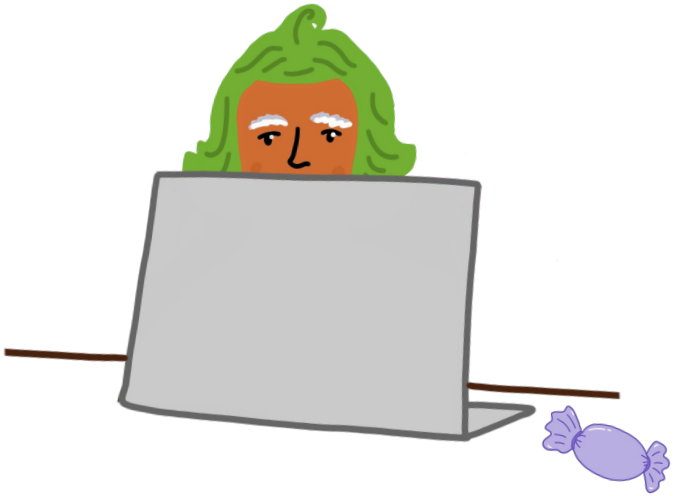
- Event IDs are Incredibly "Noisy"
- Over reliance without extra products
- Lack of actionable information
  - Encryption type (NOT session key type)
  - Ticket times
  - Flags (NOT KDC options)

Levels of access to ticket telemetry

- 1st level: ticket only
- 2nd level: ticket + the service key ← required to view the **PAC**
- 3rd level: ticket + the service key & the **krbtgt** key (DC)

More keys = more insight into a potential forged ticket

Ticket Decryption

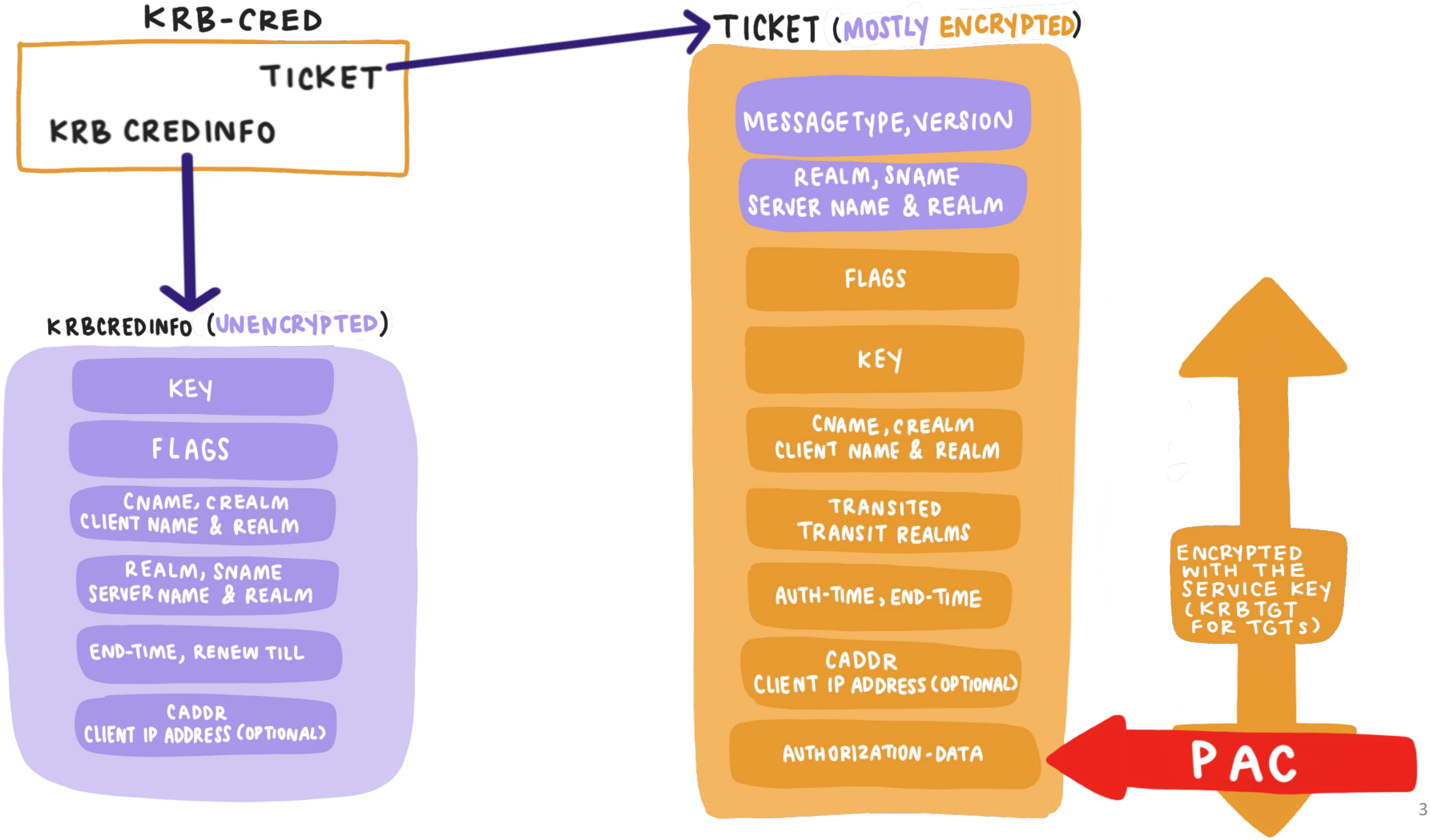


Requirement

- "On-the-fly" decryption requires keys to be DCSynced

Benefits

- Enables full ticket analysis
- Exposes addtl info to build IOAs (ex: signatures, user attributes)


KRB-CRED
TICKET
KRB CREDINFO
KRBCREDINFO (UNENCRYPTED)
KEY
FLAGS
CNAME, CREALM
CLIENT NAME & REALM
REALM, SNAME
SERVER NAME & REALM
END-TIME, RENEW TILL
CADDR
CLIENT IP ADDRESS (OPTIONAL)
TICKET (MOSTLY ENCRYPTED)
MESSAGETYPE, VERSION
REALM, SNAME
SERVER NAME & REALM
FLAGS
KEY
CNAME, CREALM
CLIENT NAME & REALM
TRANSITED
TRANSIT REALMS
AUTH-TIME, END-TIME
CADDR
CLIENT IP ADDRESS (OPTIONAL)
AUTHORIZATION-DATA
ENCRYPTED WITH THE SERVICE KEY (KRBTGT FOR TGTs)
PAC

Practical example 1 – LSA:

- Dump sessions/tickets from **LSA**
- **E**ncrypt dumps
- **DCS**ync keys for decryption and signature verification
- **Q**ueries **DC** for proper information to perform analysis

WonkaVision: A tool to Detect Forged Tickets

Initial Approach

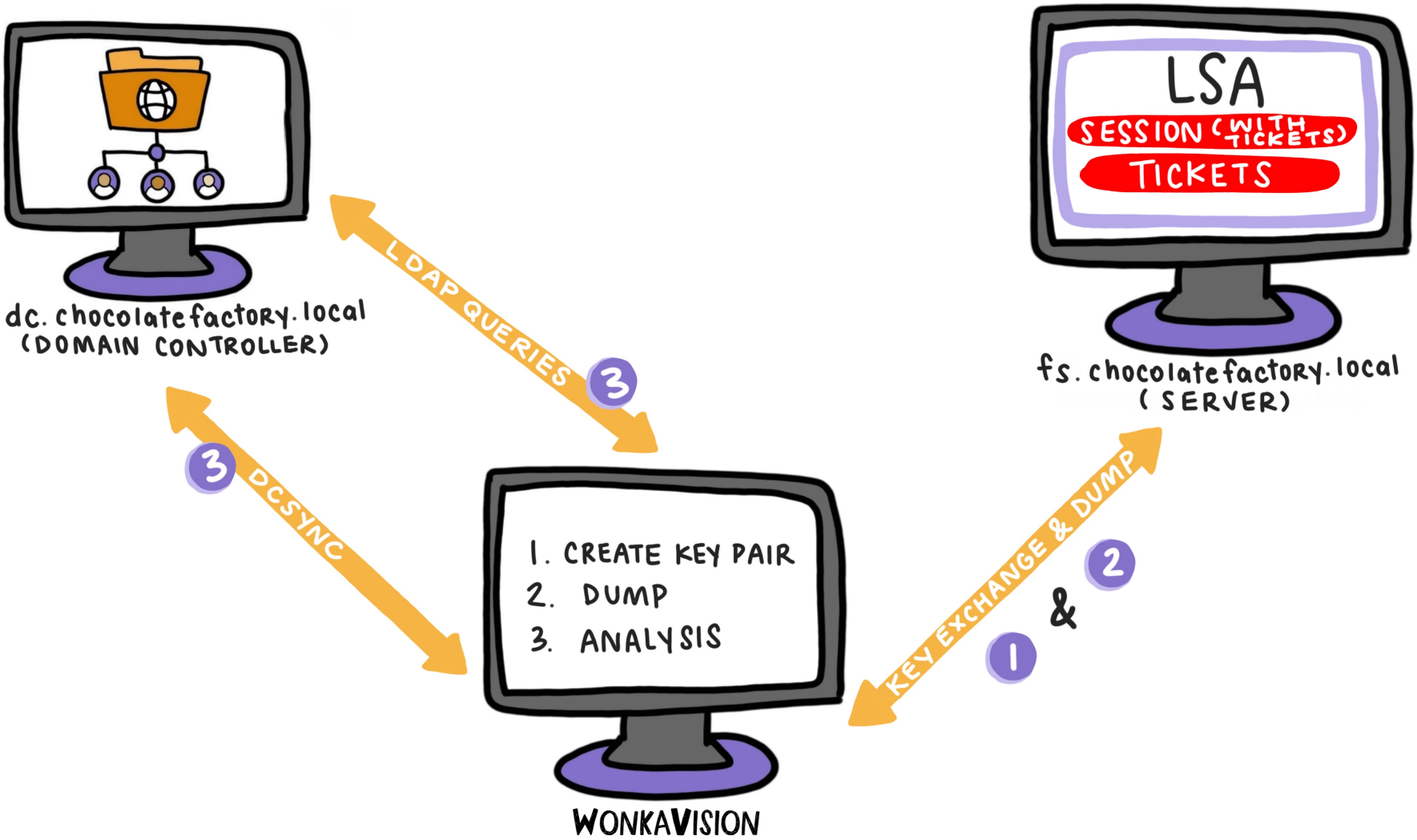
- Utilized "capability abstraction" with known **OST**s

Execution

- Attack simulation and compare against "known good"

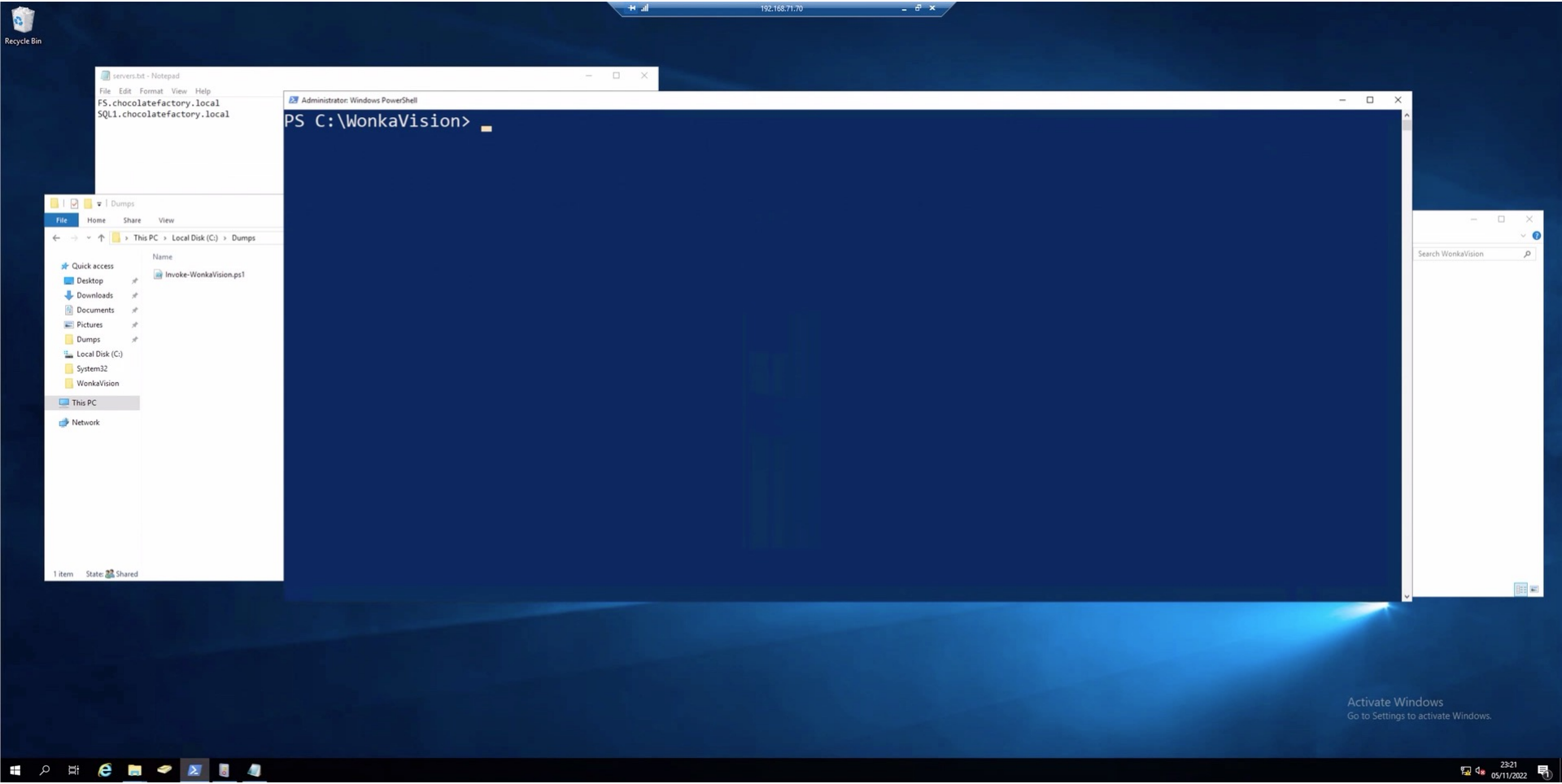
Outcome

- Improved detection capabilities w/ scoring **(IOAS)** based on **OST** signatures


dc.chocolatefactory.local
(DOMAIN CONTROLLER)
LDAP QUERIES
3
LSA
SESSION (WITH TICKETS)
TICKETS
fs.chocolatefactory.local
(SERVER)
3
DCSYNC
1. CREATE KEY PAIR
2. DUMP
3. ANALYSIS
KEY EXCHANGE & DUMP
2
&
1
WONKAVISION

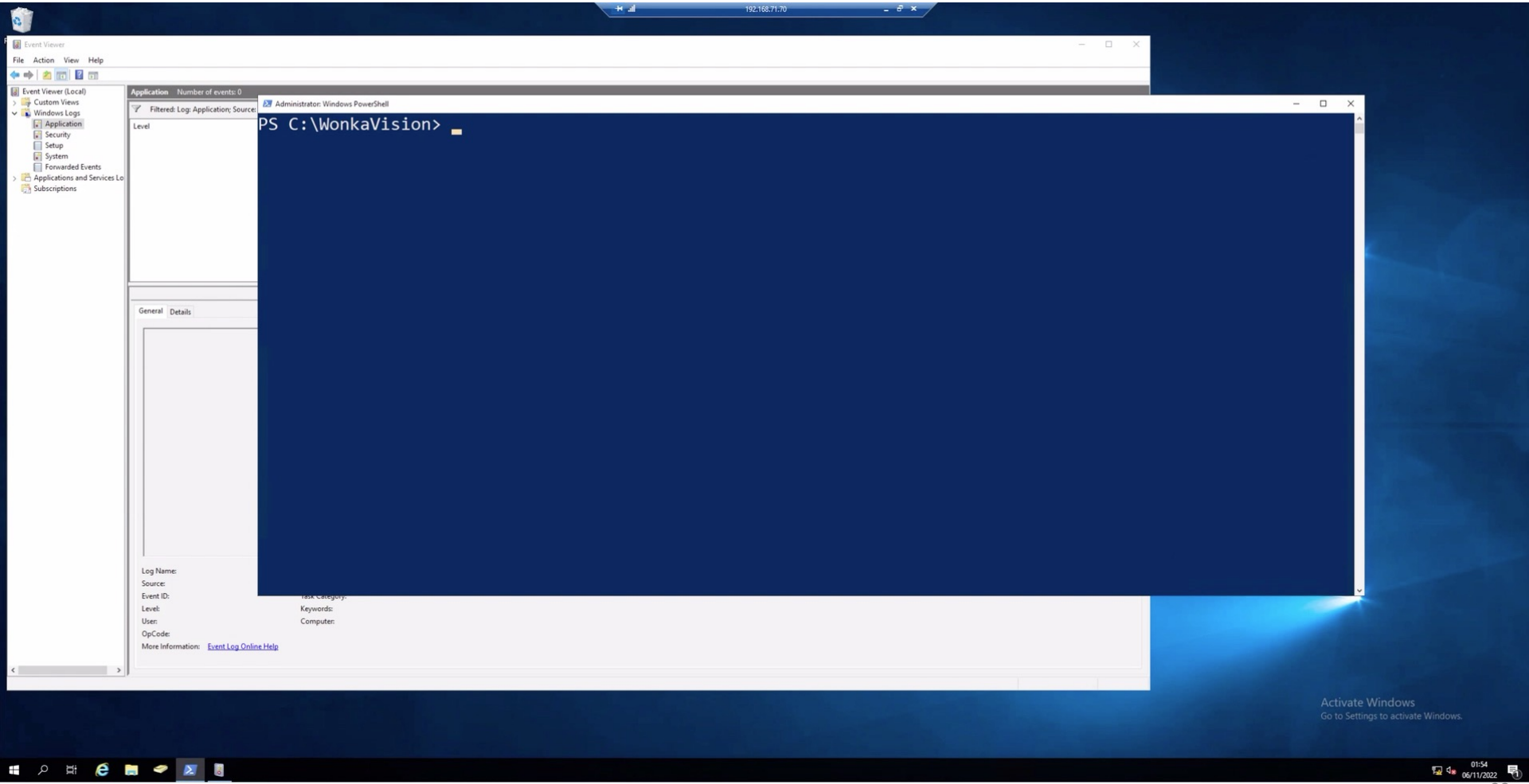
DEMO

DEMO 2: WV KEY CREATION & SESSION/DUMPING COMMANDS

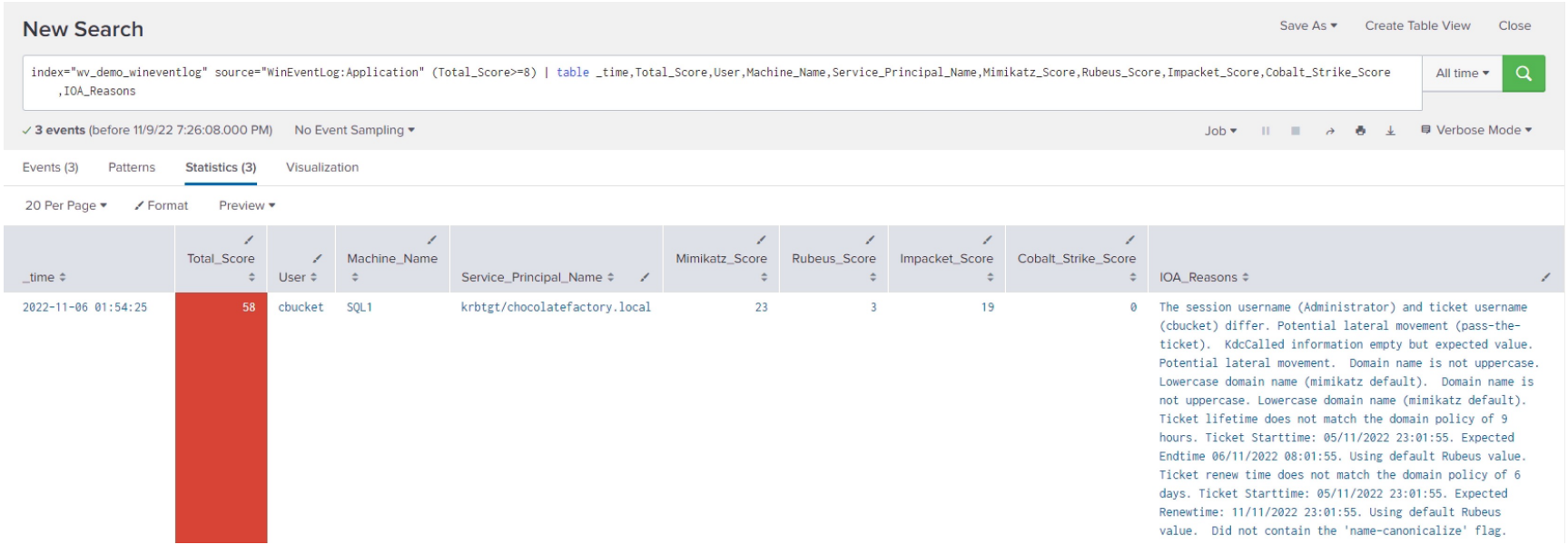

Recycle Bin
192.168.71.70
servers.txt - Notepad
File Edit Format View Help
FS.chocolatefactory.local
SQL1.chocolatefactory.local
Administrator: Windows PowerShell
PS C:\WonkaVision>
Dumps
File Home Share View
This PC › Local Disk (C:) › Dumps
Quick access
Desktop
Downloads
Documents
Pictures
Dumps
Local Disk (C:)
System32
WonkaVision
This PC
Network
Name
Invoke-WonkaVision.ps1
1 item State: Shared
Search WonkaVision
Activate Windows
Go to Settings to activate Windows.
23:21
05/11/2022

DEMO

DEMO 3: ANALYSIS COMMAND & OUTPUT IN EVTX


192.168.71.70
Event Viewer
File Action View Help
Event Viewer (Local)
Custom Views
Windows Logs
Application
Security
Setup
System
Forwarded Events
Applications and Services Lo
Subscriptions
Application Number of events: 0
Filtered: Log: Application; Source:
Level
General Details
Log Name:
Source:
Event ID:
Level:
User:
OpCode:
More Information: Event Log Online Help
Keywords:
Computer:
Administrator: Windows PowerShell
PS C:\WonkaVision>
Activate Windows
Go to Settings to activate Windows.
01:54
06/11/2022

SIEM LOG FORWARDING - SPLUNK



SIEM LOG FORWARDING - SENTINEL

TimeGenerated [UTC]	ParsedEventData	Source	EventLog	Computer	EventLevel	EventLevelNam

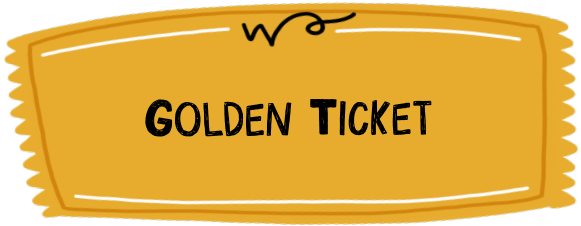
ParsedEventData	{"DataItem":{"@type":"System.XmlData","@time":"2022-11-05T13:39:39.5008352Z","@sourceHealthServiceId":"b9237f0b-b5e5-ba00-0951-44654bfd7af6","EventData":{"@xmlns":"http://schemas.micr
TotalScore	17
Session	0x156af5
MachineName	Asgard-Wrkstn
User	thanos
ServicePrincipalName	krbtgt/marvel.local
IOAs	
IOA_SessionUser	thor
IOA_KDCCalled	TicketFlags: pre_authent, initial, renewable, forwardable UpnDNSBuffer: Not Extended RequestorBuffer: None AttributesBuffer: None Tool Scores:
TScore_MimikatzScore	4
TScore_ImpacketScore	4
TScore_RubeusScore	2
TScore_CobaltStrikeScore	0
IOA_Reasons	The session username (thor) and ticket username (thanos) differ. Potential lateral movement (pass-the-ticket). KdcCalled information empty but expected value. Potential lateral movement. Did not c

THANKS TO JONATHAN JOHNSON (@JSECURITY101) FOR PROVIDING THIS SCREENSHOT

WONKAVISION: PERSONAS & USE CASES

SOC ANALYST

NOTABLE ALERTING



PROACTIVE HUNTING

TARGETED SAMPLES
RANDOM SAMPLES

IR & REACTIVE HUNTING

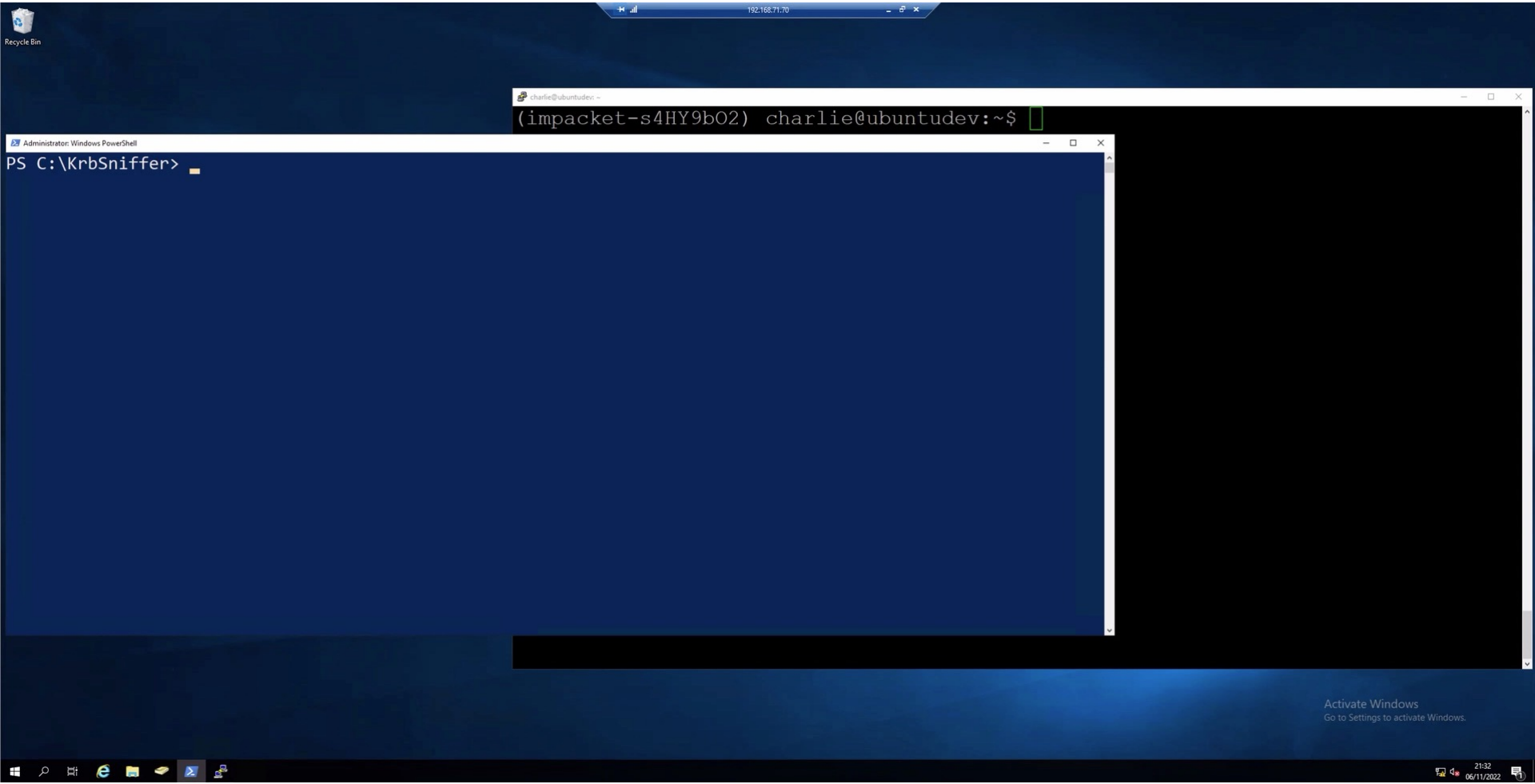
EVIDENCE OF DCSYNC

Practical example 2 – network:

- Captures traffic sent to DCs
- Fully decodes **TGS-REQ**s off the network
- Performs basic analysis of unencrypted data
- Could easily decrypt on-the-fly & do full analysis

DEMO

DEMO 4: KERBEROS SNIFFER ANALYZING POSSIBLE GT USAGE


Recycle Bin
192.168.71.70
charlie@ubuntudev: ~
(impacket-s4HY9bO2) charlie@ubuntudev:~$
Administrator: Windows PowerShell
PS C:\KrbSniffer>
Activate Windows
Go to Settings to activate Windows.
21:32
06/11/2022

LSA vs. Network

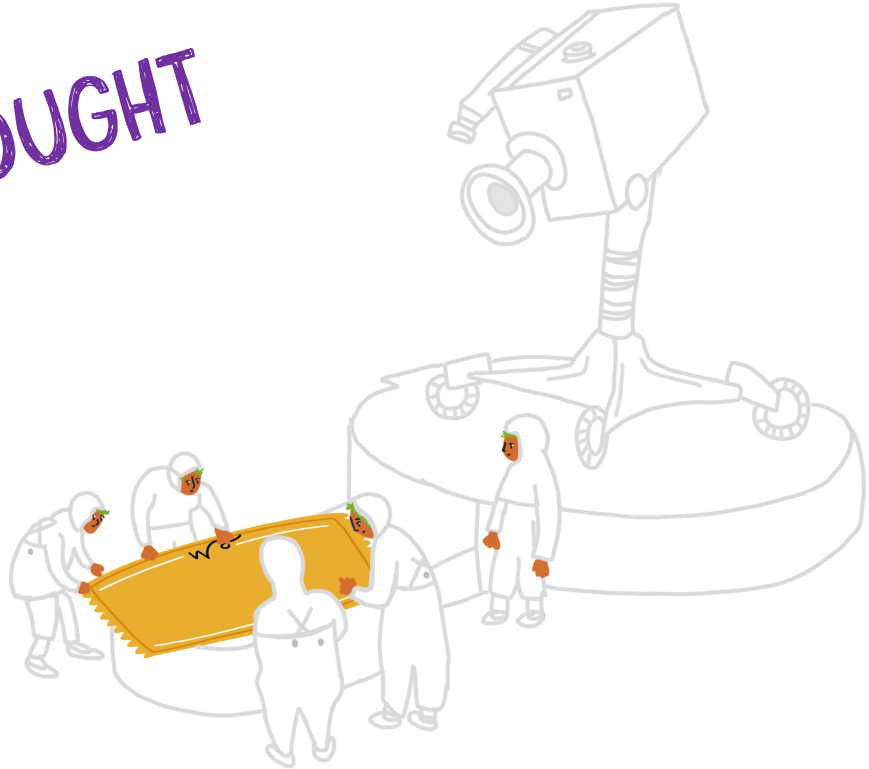
LSA

- Returns more unencrypted info (e.g. ticket flags & ticket times)
- Shows session info
- Sees service tickets & TGTs w/o passing many different protocols

Network

- Can use info about the TGS-REQ vs. just the ticket
- TGS-REQs can all be viewed looking at only DC network traffic
- Will see TGS-REQs when LSASS isn't being used (Rubeus, impacket, Kekeo)

Final Thought



There's no guarantee you will detect a forged ticket, but decrypting on-the-fly increases your chance of success

QUESTIONS?