# BREAKING THE ATTACK GRAPH:

# HOW TO LEVERAGE GRAPHS TO STRENGTHEN SECURITY IN A DOMAIN ENVIRONMENT

MARINA SIMAKOV

Microsoft

black hat ASIA 2018

# AGENDA

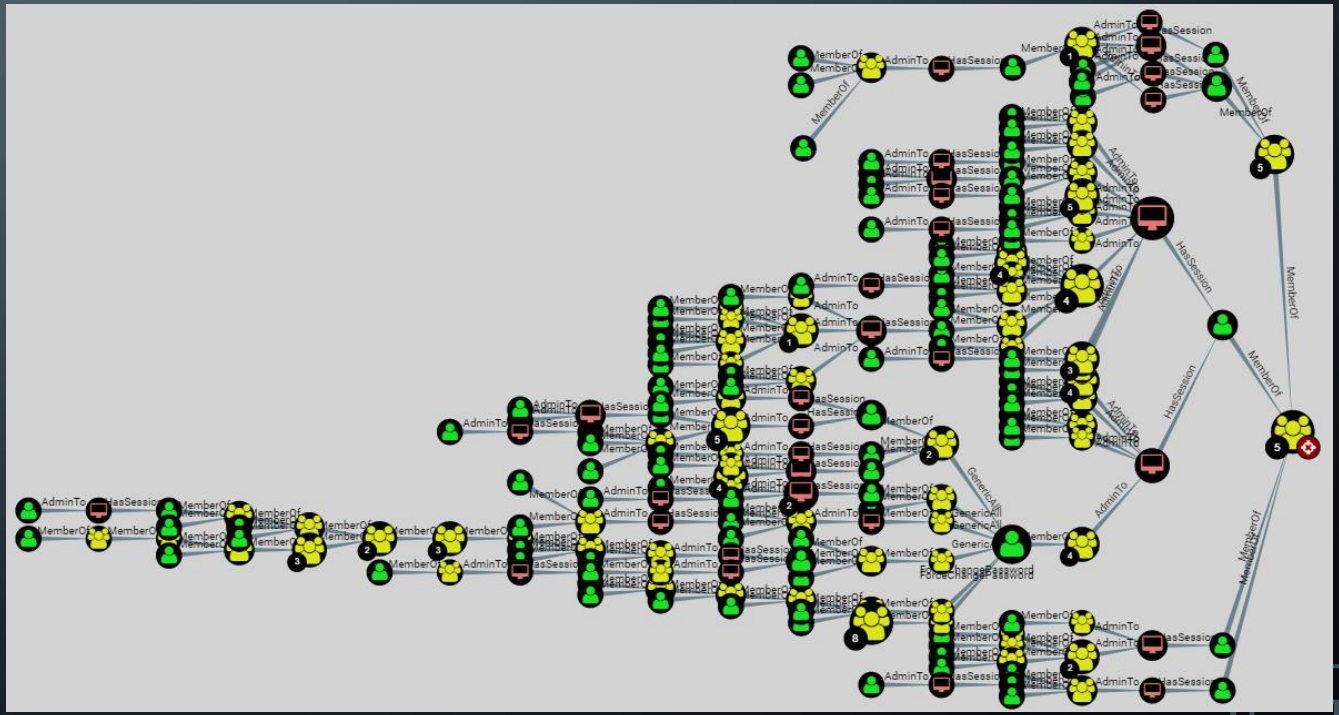
- How attack tools use graphs & their limitations
  - BloodHound
  - GoFetch
- How defenders can use graphs
  - Prevention: reduce attack surface
  - Detection: reconnaissance & lateral movement
  - Investigation

# BLOODHOUND

- Scans the network:
  - Local Administrators
  - Domain group memberships
  - Active sessions
  - ACLs

- Finds shortest paths to domain admins

https://github.com/BloodHoundAD/BloodHound
@_wald0, @CptJesus, and @harmj0y



# GoFetch

- Automates the lateral movement process
- Input:
  - A path generated by BloodHound
- Output:
  - Domain admin credentials
- The length of the path is not a factor


https://github.com/GoFetchAD/GoFetch
@TalTheMaor

# ATTACKERS VS. DEFENDERS

| | ATTACKERS | DEFENDERS |
|---|---|---|
| Permissions (data gathering) | Limited (domain user) | Unlimited |
| Access to network resources | Limited (dependent on env & hardening) | Unlimited |
| Graph | Partial | Full |
| Result | Valuable insights & Complex attack paths |  |

# HOW CAN DEFENDERS USE GRAPHS?

Prevention → Detection → Investigation

- Detect vulnerable nodes:
  - Users
  - Computers
- Disconnect them from as many attack paths as possible

- Study logon patterns
- Detect anomalies:
  - Reconnaissance
  - Lateral movement

- Discover the attack path
- Find additional compromised machines\users

# DATA SOURCES

1. Domain group memberships
2. Local group memberships on domain machines
3. Existing credentials on each machine (NTLM \ Kerberos)
   - Logon events (SIEM \ DC)
   - Network traffic to DC (successful AS & TGS requests)
   - No need to constantly query machines
   - Use a single data source

# PREVENTION

# DETECTION

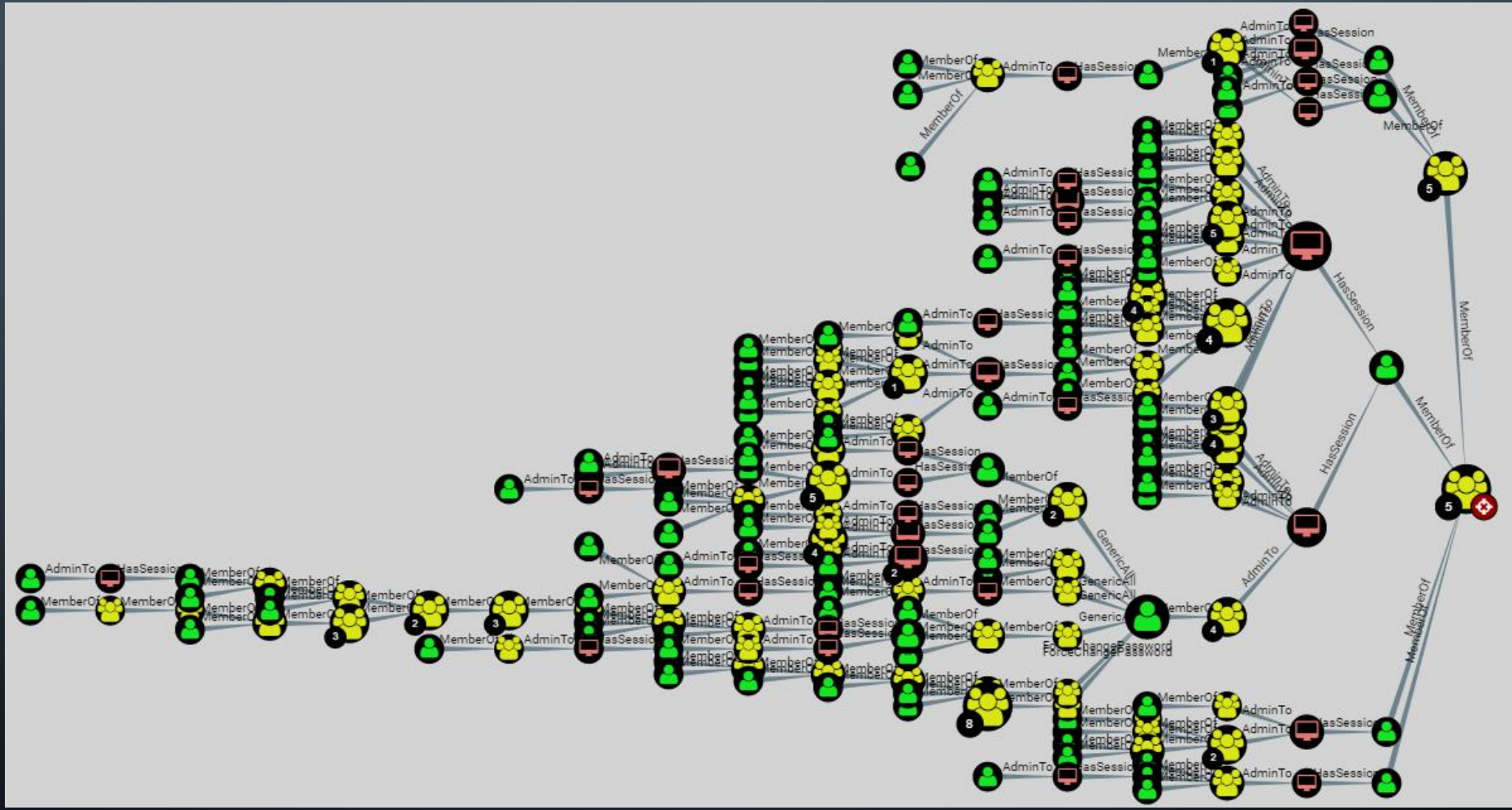
# INVESTIGATION

# PREVENTION

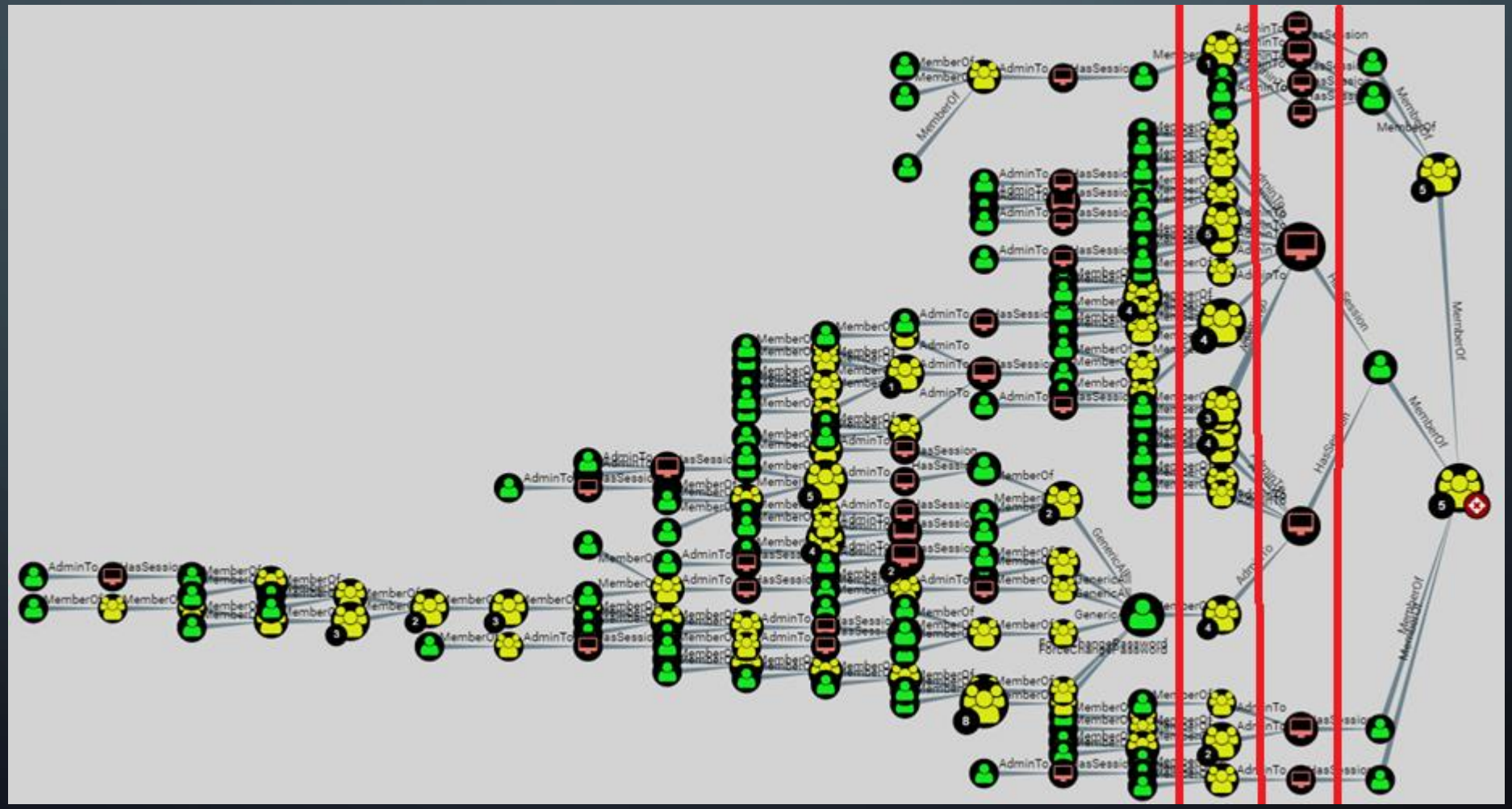
- **Goal:** reduce attack surface
- **How:**
  - Discover **vulnerable nodes** in the network
  - Disconnect them from as many attack paths as possible

- **Problem:** Computing all the paths is hard
  - Running time
  - Resources
  - Constantly changing
- **Solution:**
  - Compute only the **last layers**
  - Disconnect one of the last edges in the paths

# PREVENTION

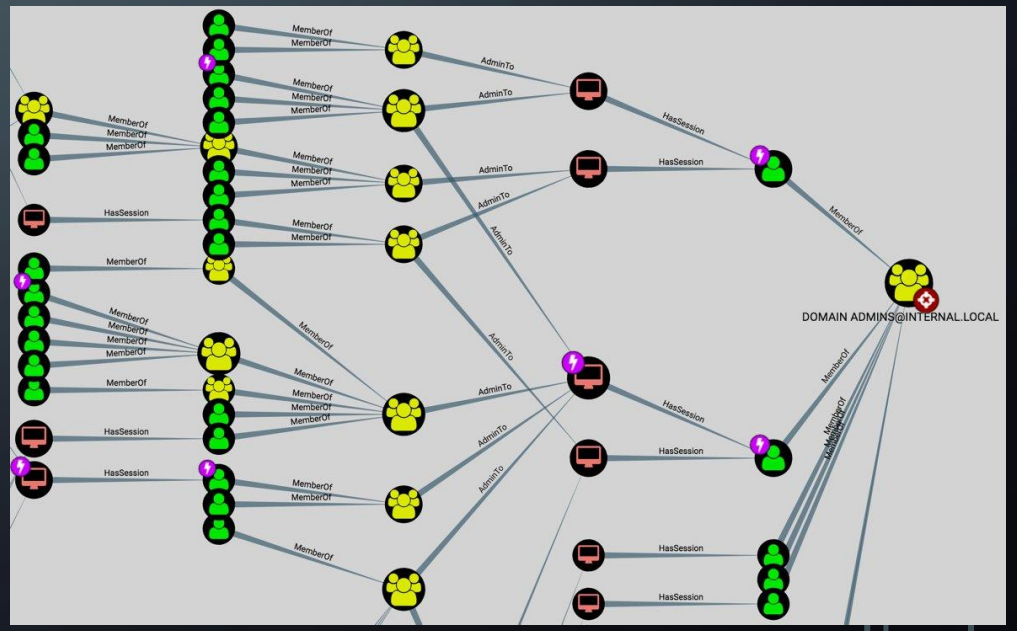

AdminTo
HasSession
MemberOf
GenericAll
ForceChangePassword

# PREVENTION


AdminTo
HasSession
MemberOf
GenericAll
ForceChangePassword

# PREVENTION: VULNERABLE NODES

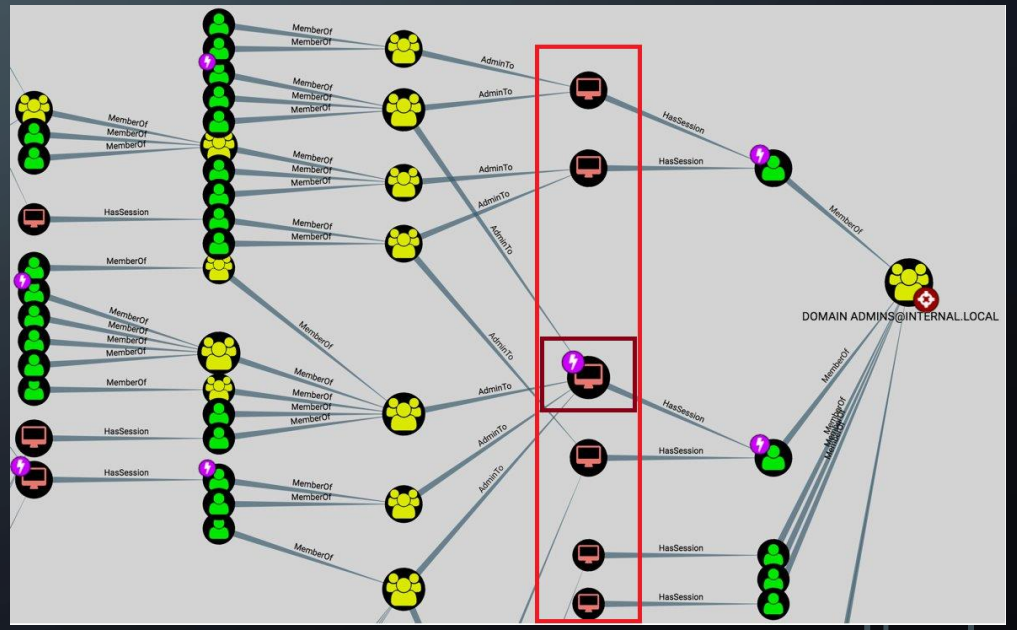
- **<u>Vulnerable node:</u>** a node which has a high probability of being compromised during an attack



# PREVENTION: VULNERABLE NODES

- **<u>Vulnerable node:</u>** a node which has a high probability of being compromised during an attack

- **<u>Vulnerable Computer:</u>**
  - Has a session of a high privileged account
  - Has many low privileged accounts with local admin privileges

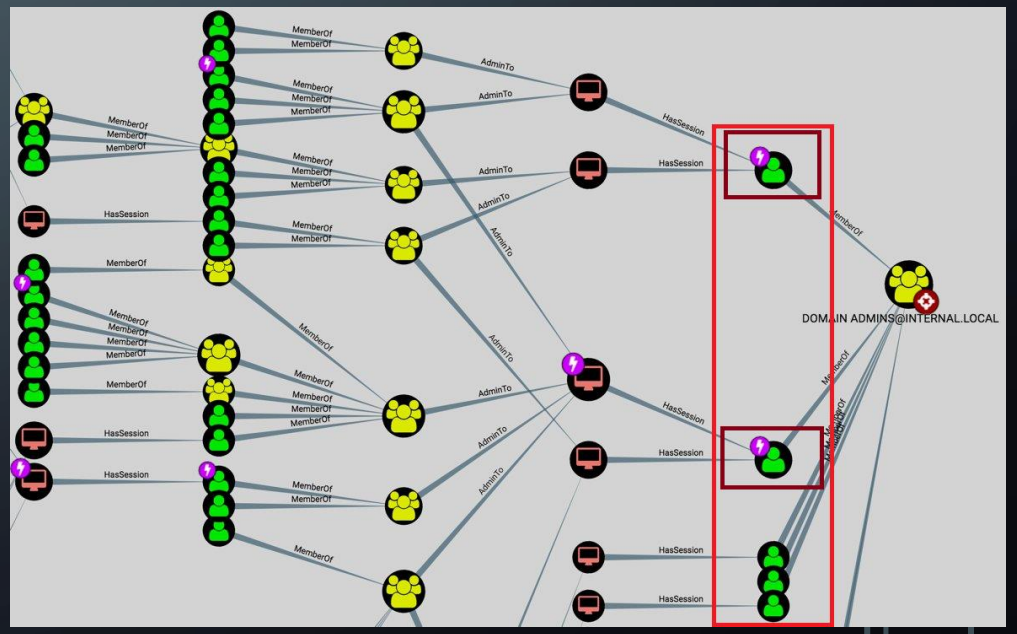


# PREVENTION: VULNERABLE NODES

- **Vulnerable node:** a node which has a high probability of being compromised during an attack

- **Vulnerable Computer:**
  - Has a session of a high privileged account
  - Has many low privileged accounts with local admin privileges

- **Vulnerable User:**
  - A high privileged account
  - Has sessions on vulnerable computers



# PREVENTION: VULNERABLE NODES

- The status of a node can change based on:
  - User behavior
  - Computer configuration
- Track the status of nodes over time
  - What percentage of the time is the node considered vulnerable?
  - Constant issue or a one-time occurrence?
  - Rank the nodes by vulnerability
- Secure the most vulnerable nodes
  - Deception – traps (Tom Sela, @4x6hw https://www.youtube.com/watch?v=elf8NK1GR-M)
  - Disconnect the vulnerable nodes from the attack graph

# PREVENTION: DISCONNECTING NODES

| | Configuration changes |
|---|---|
| **How?** | • Examine local admins and their logon patterns<br>• Remove redundant members |
| **When?** | • Machines with many high privileged sessions<br>• Inactive local admins |

# PREVENTION: DISCONNECTING NODES

| | Configuration changes | Behavioral changes |
|---|---|---|
| **How?** | • Examine local admins and their logon patterns<br>• Remove redundant members | • Network logon<br>• Logon to vulnerable machine using a less privileged account<br>• Remote Cred-Guard<br>• Log off |
| **When?** | • Machines with many high privileged sessions<br>• Inactive local admins | • Ideally: always ☺ |

# PREVENTION: DISCONNECTING NODES

| | Configuration changes | Behavioral changes | Active monitoring* |
|---|---|---|---|
| **How?** | • Examine local admins and their logon patterns<br>• Remove redundant members | • Network logon<br>• Logon to vulnerable machine using a less privileged account<br>• Remote Cred-Guard<br>• Log off | • Disconnect inactive sessions of high privileged accounts |
| **When?** | • Machines with many high privileged sessions<br>• Inactive local admins | • Ideally: always ☺ | • A security solution with admin privileges on the machine |

➡ We can significantly reduce the attack surface.
However, we cannot remove it entirely.

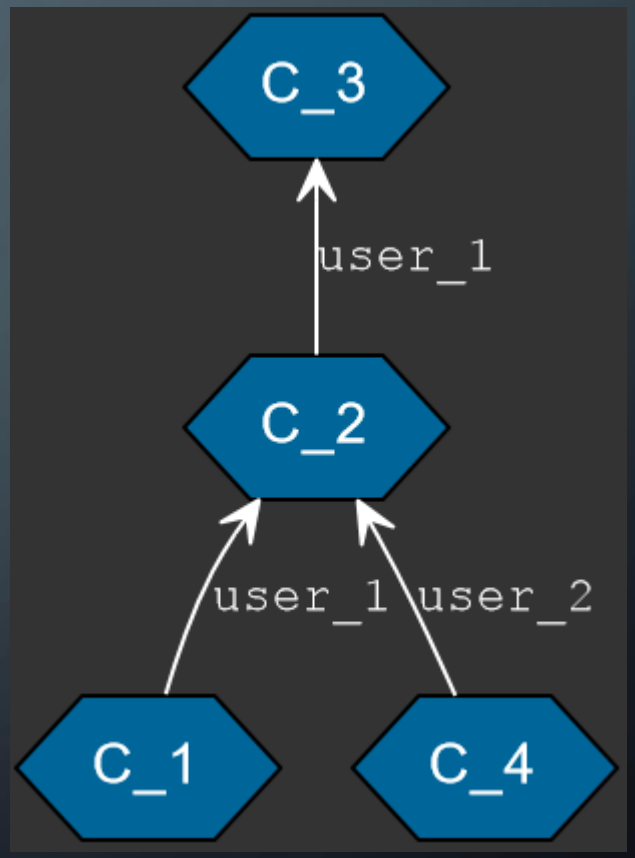
PREVENTION

**DETECTION**

INVESTIGATION

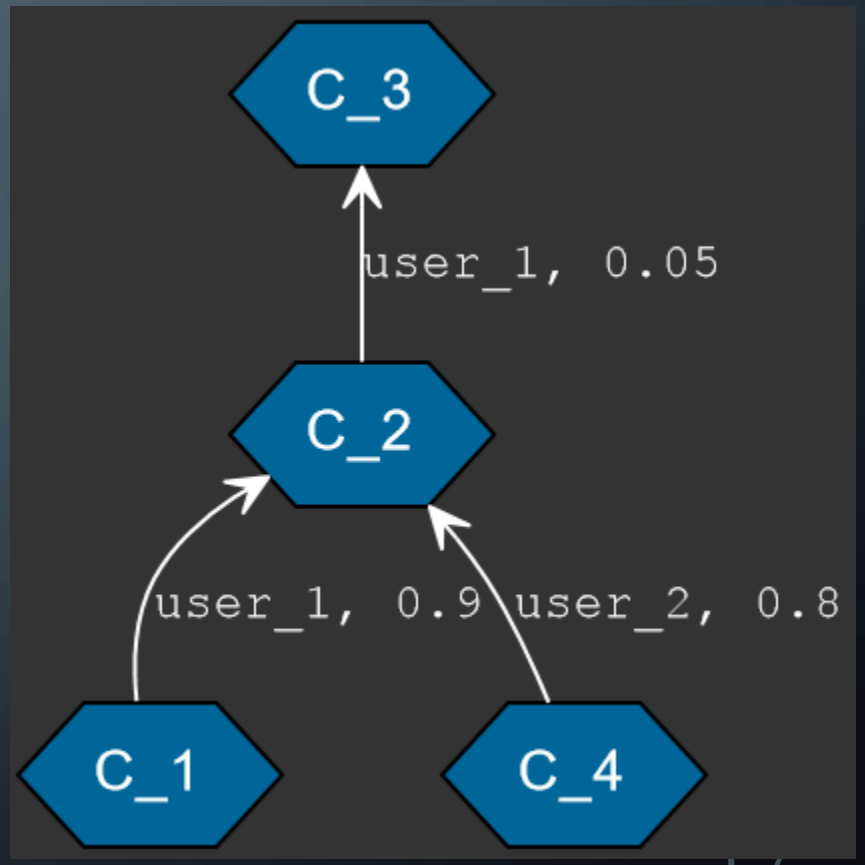
# DETECTION: LOGON GRAPH

- Logon Graph $G = (V, E)$:
  - $V = \{v_1, v_2, \dots, v_n)$: Domain computers
  - $E = \{e_1, e_2, \dots, e_m\}$: Sessions
  - Each edge has a label
  - $l(e_i = (v_k, v_l)) = u_j \mid$
    $u_j$ *is the domain user which connected from* $v_k$ *to* $v_l$



# DETECTION: WEIGHTED LOGON GRAPH

- Weighted Logon Graph $G = (V, E, w)$:
  - $V = \{v_1, v_2, \dots, v_n)$: Domain computers
  - $E = \{e_1, e_2, \dots, e_m\}$: Sessions
  - Each edge has a label & a weight
  - $l(e_i = (v_k, v_l)) = u_j \mid$ $u_j$ *is the domain user which connected from* $v_k$ *to* $v_l$
  - $w(e_i = (v_k, v_l)) = P(user\ u_j\ connecting\ from\ v_k\ to\ v_l)$



# DETECTION

1. **Weighted Logon Graph:** constructed based on logon activities during a learning period
   - An edge $e = (v_k, v_l)$ is assigned a weight based on the logon activities of the corresponding user on $v_k, v_l$

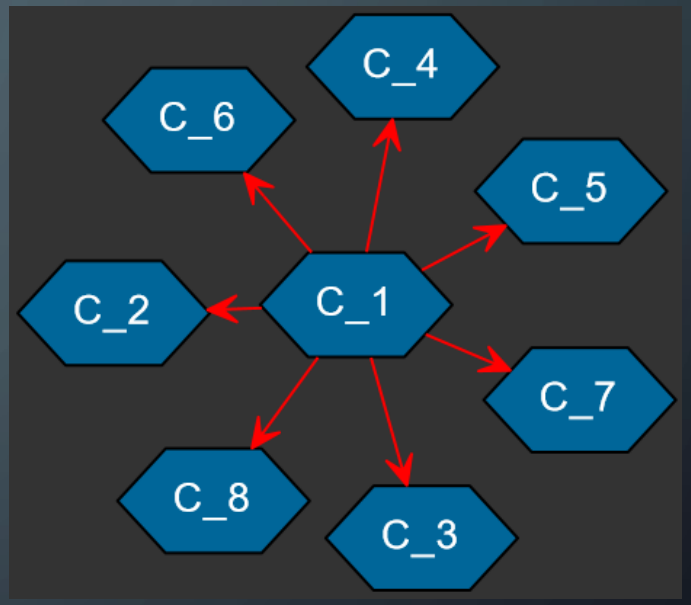
2. **Daily Logon Graph:** constructed based on logon activities during the day and the Weighted Logon Graph
   - Construct a logon graph based on the daily logon activities
   - Delete edges which are assigned a high weight in the Weighted Graph

3. **Detect anomalies** on the Daily Logon Graph

# DETECTION

Detecting anomalies:

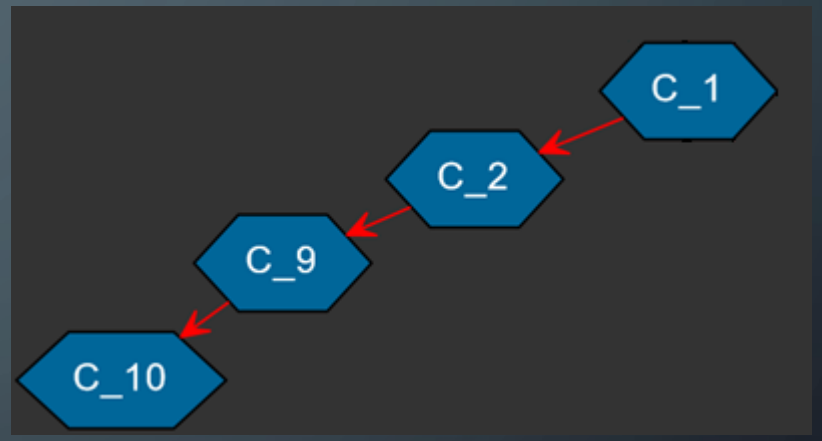
- Star structures
  - Reconnaissance activities
  - Main attack machine
- Chains
  - Lateral movement
- Combination



# DETECTION

Detecting anomalies:

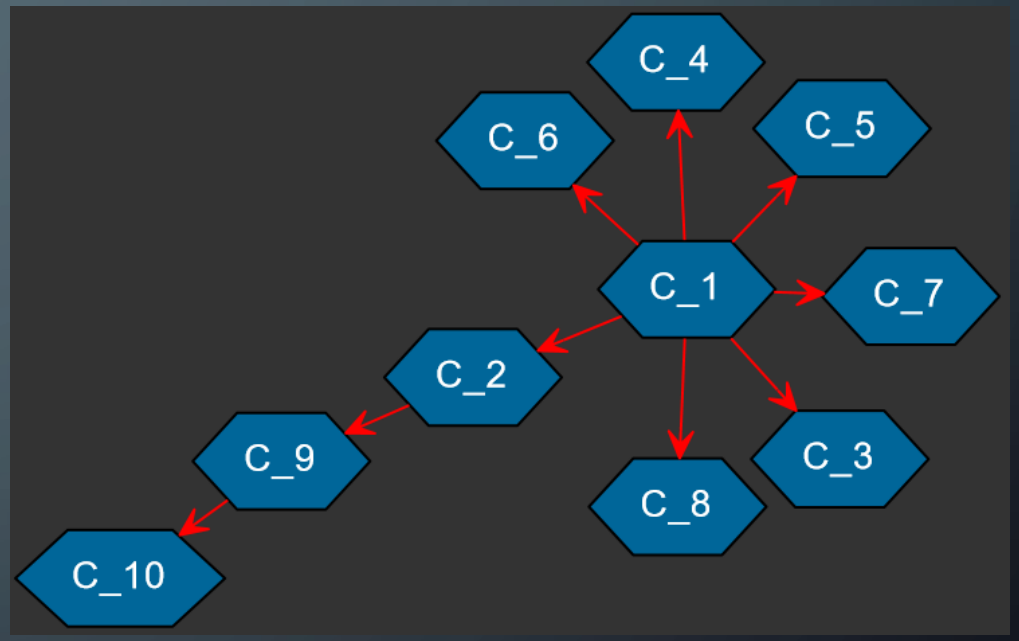
- Star structures
  - Reconnaissance activities
  - Main attack machine
- Chains
  - Lateral movement
- Combination



# DETECTION

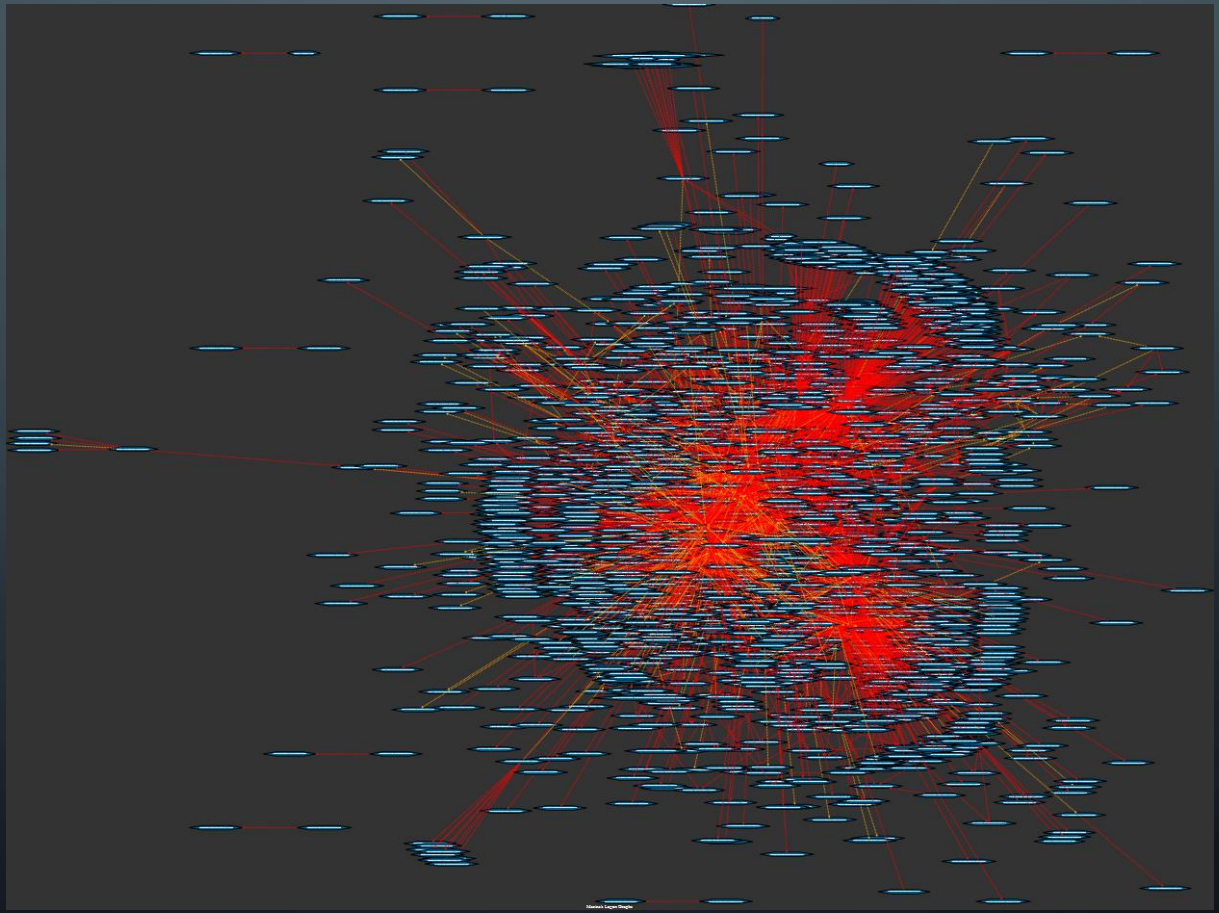
Detecting anomalies:

- Star structures
  - Reconnaissance activities
  - Main attack machine
- Chains
  - Lateral movement
- Combination



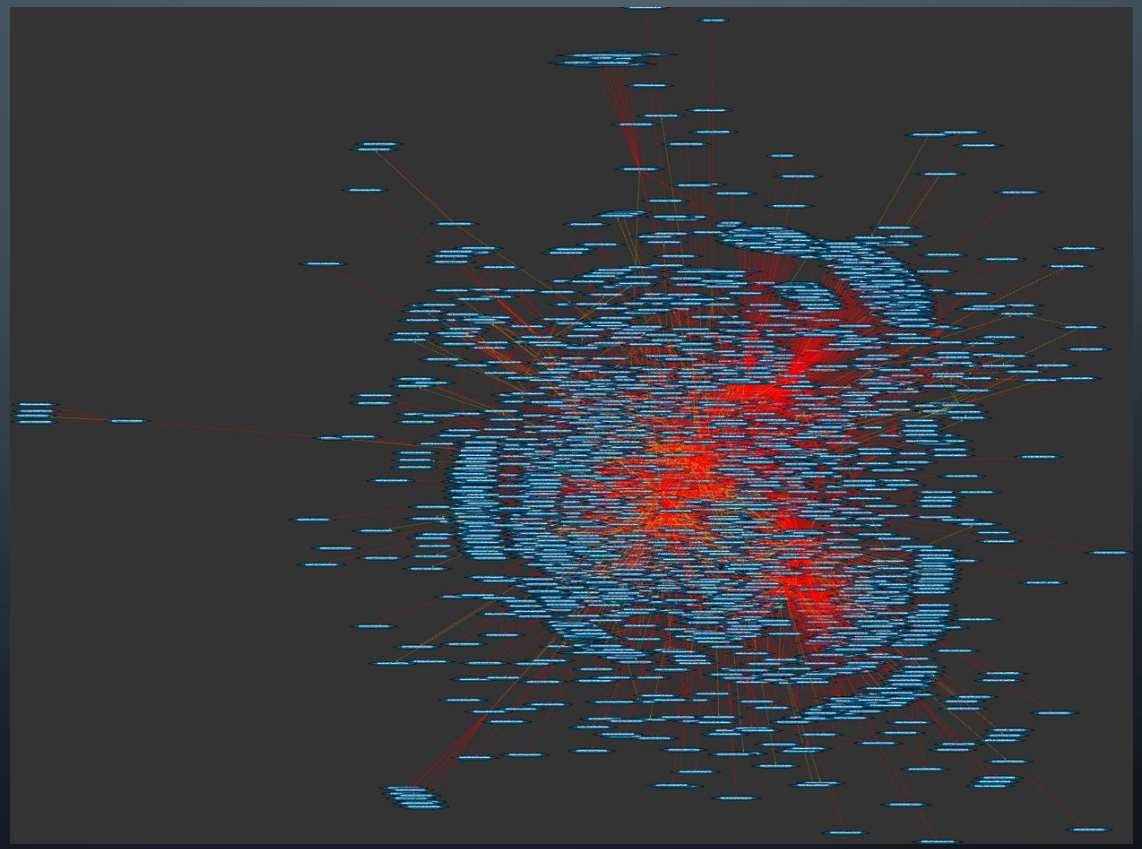
# DETECTION: EXAMPLE

- **<u>Try 1:</u>** Consider all suspicious edges



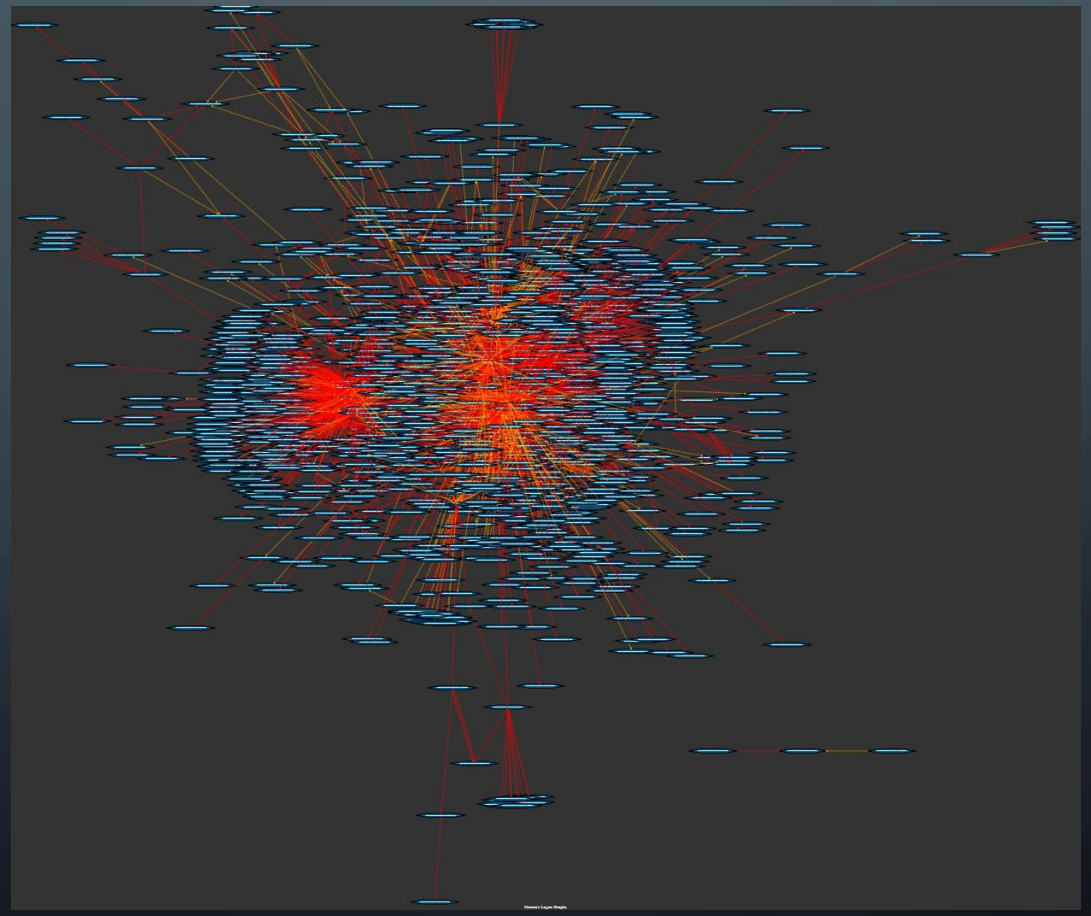
# DETECTION: EXAMPLE

- **<u>Try 2:</u>** Remove connected components consisting of 2 nodes (2 nodes connected by an edge without additional connections)



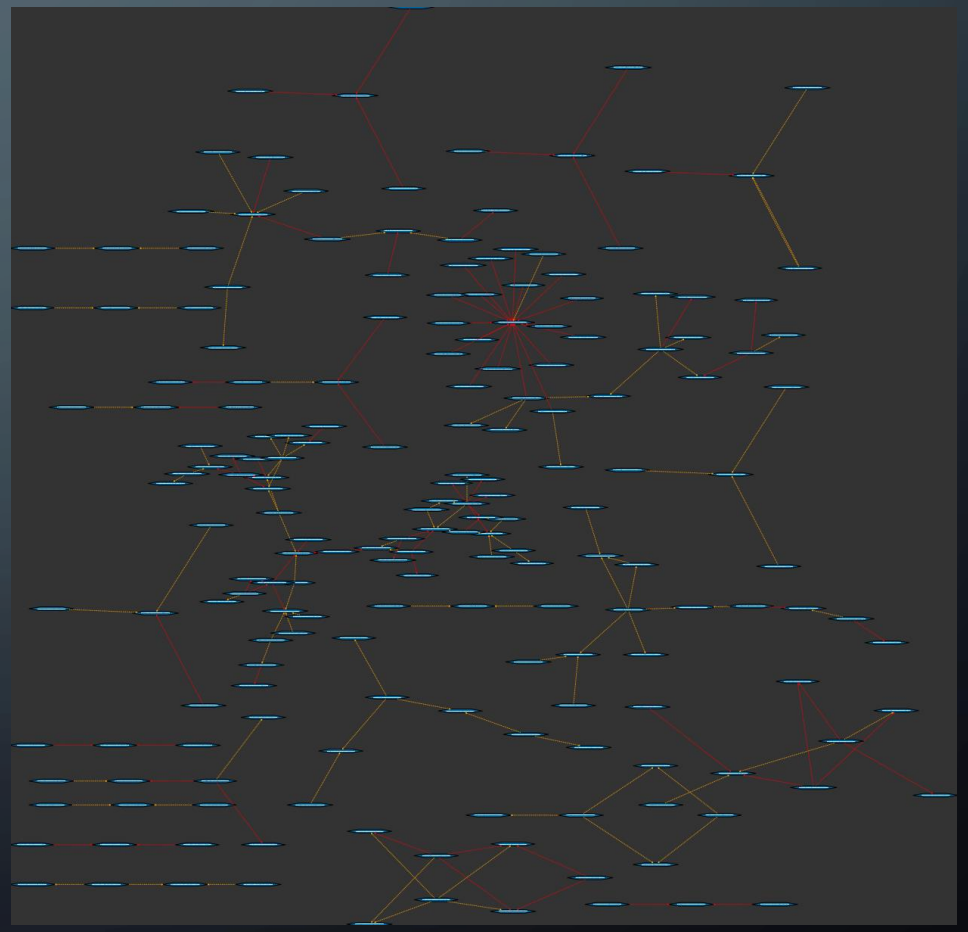
# DETECTION: EXAMPLE

- **<u>Try 3:</u>** Separate “stars” structures from “chains”



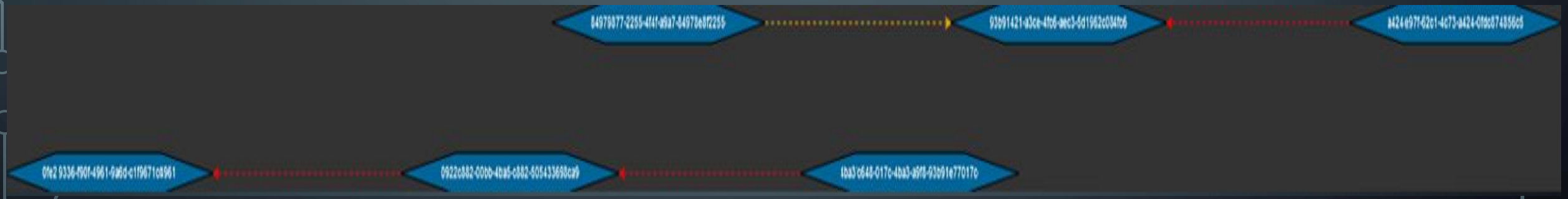
# DETECTION: EXAMPLE

- **Try 4:** Add some more interesting features…
  - New users & machines
  - Computer & resource popularity
  - Peers behavior



# DETECTION: EXAMPLE

- **<u>Try 5:</u>** Consider connections where the user is a local admin on the target machine
- Result: Not too bad ☺



# DETECTION: PROS & CONS

| PROS | CONS |
|---|---|
| In order to fully avoid detection, attackers need to be aware of the entities' behavior | If the attack corresponds to an entity's usual behavior, it would be harder to detect |
| Detect paths involving **multiple users** | **Weight threshold:**<br>Too high: False Positives<br>Too low: might miss real attacks |
| Detect outsider & insider threats | Hard to detect "scattered attacks" |
| Can be incorporated with endpoint data to strengthen certainty & eliminate FPs | |

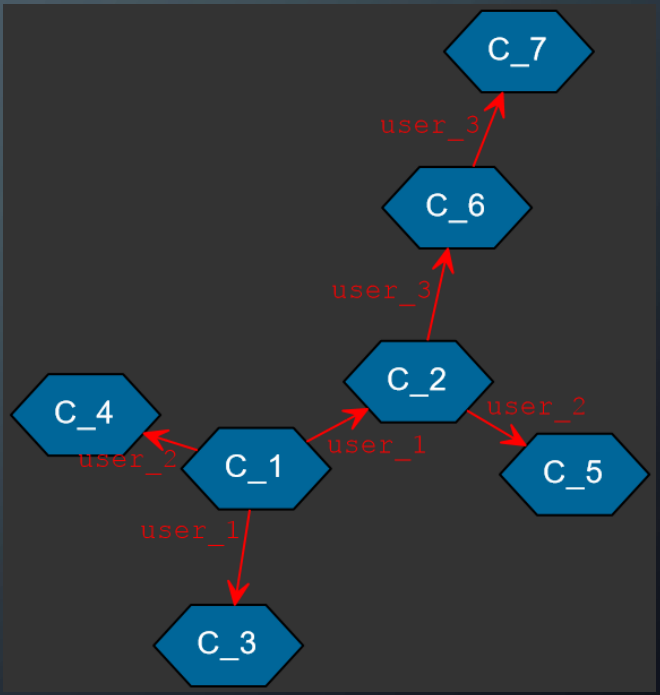
PREVENTION

DETECTION

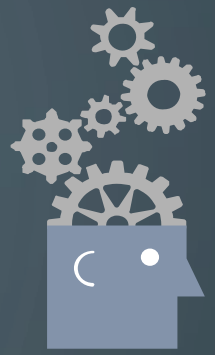
INVESTIGATION

# INVESTIGATION

1. Input:
   - Known compromised machines
   - Known compromised users
2. Analysis:
   - Construct connection graphs for known compromised entities
3. Output:
   - Information about the attack path
   - Additional accounts which may have been compromised



# CONCLUSIONS



- Attackers have access to limited data sources
- However, are still able to leverage graphs to gain insights into the environment & compromise high privileged accounts

- Defenders have access to unlimited data sources
- This data can be used to create very informative graphs to aid in:
  - Prevention
  - Detection
  - Investigation

- Graphs are awesome

# THANK YOU ☺