Truth and Consequences

December 6, 2015

Eugene H. Spafford

Working with
Jeff Avery
Chris Gutierrez
Prof. Mohammed Almeshekah
Prof. Saurabh Bagchi

The “Liars Club”

PURDUE
UNIVERSITY®

CERIAS®
Center for Education and Research
in Information Assurance and Security

What People Accept As True…

Think about this “truism” of cyber security:

There is no security through obscurity.

Makes sense, right?

What People Accept As True…

What People Accept As True…

No, it doesn’t! Try publishing your password. Expose your encryption keys. Make your firewall config world readable.

In actual practice, obscurity **adds** to security!

What People Accept As True…

No, it doesn’t! Try publishing your password. Expose your encryption keys. Make your firewall config world readable.

In actual practice, obscurity **adds** to security!

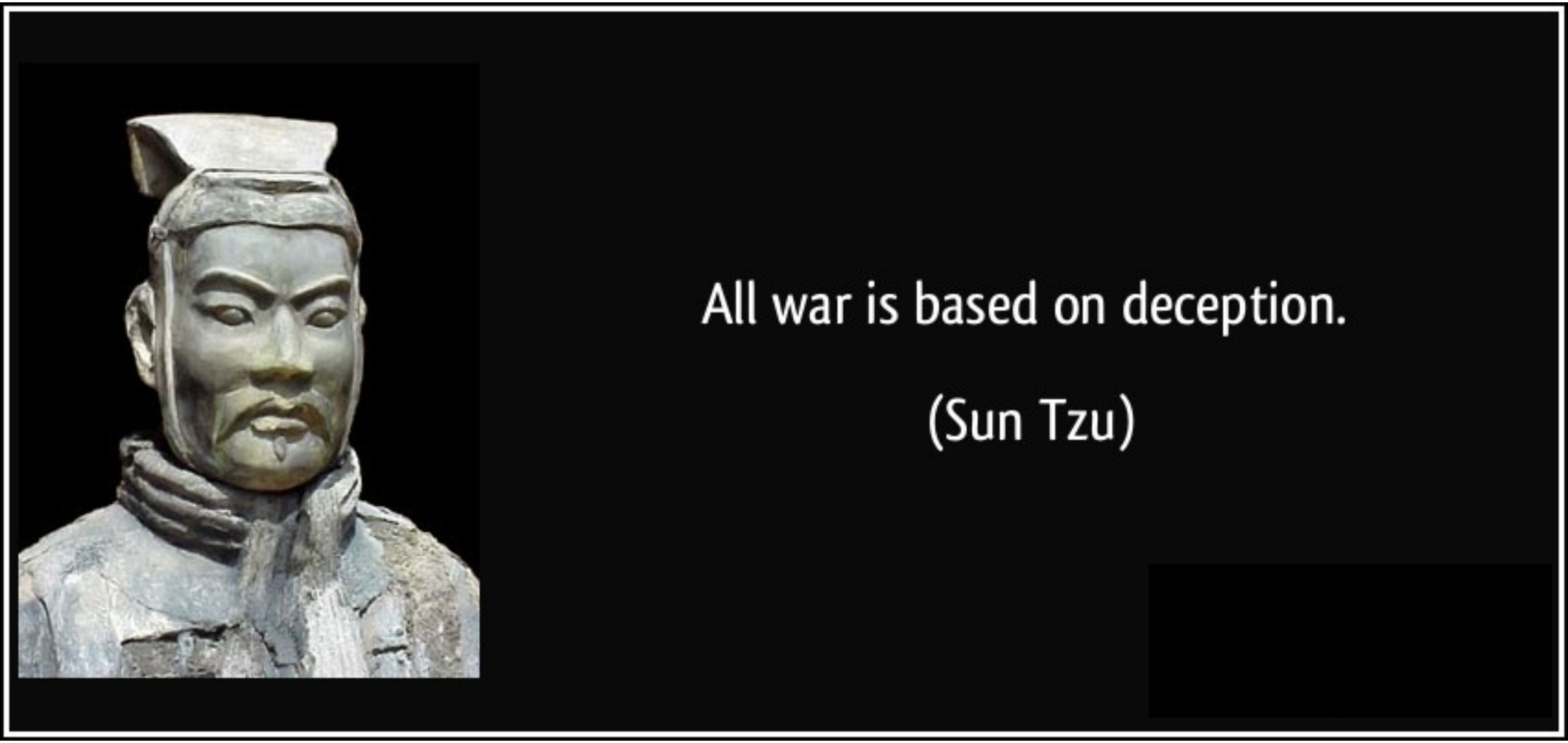
What People Accept As True…

No, it doesn’t! Try publishing your password. Expose your encryption keys. Make your firewall config world readable.

In actual practice, obscurity **adds** to security!

This “truism” is a corruption of Kerckhoff’s Principle:

*A **cryptosystem** should be secure even if everything about the system, except the key, is public knowledge.*

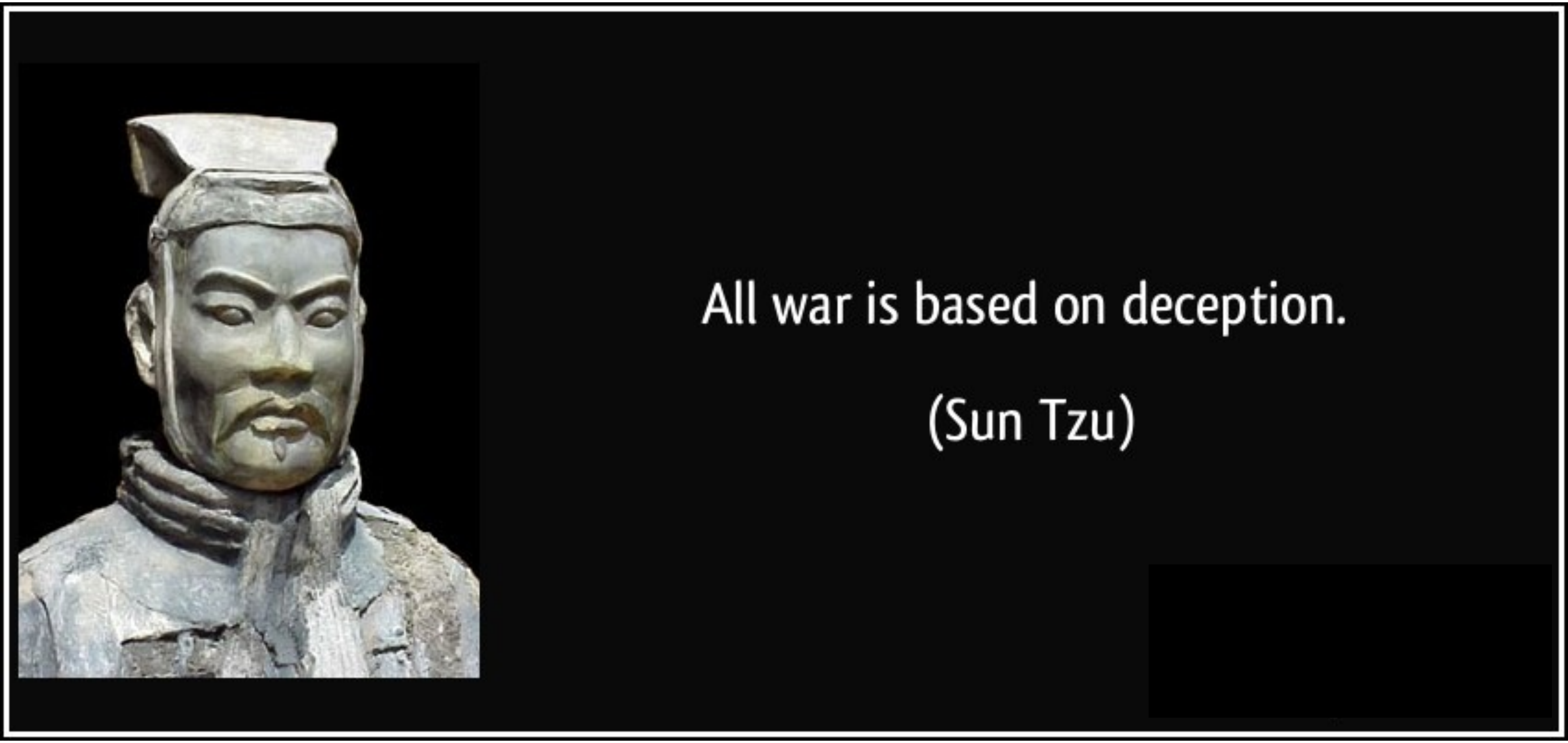


All war is based on deception.

(Sun Tzu)

[redacted]

Deception in conflict is an old, tried-and-true concept!



All war is based on deception.

(Sun Tzu)

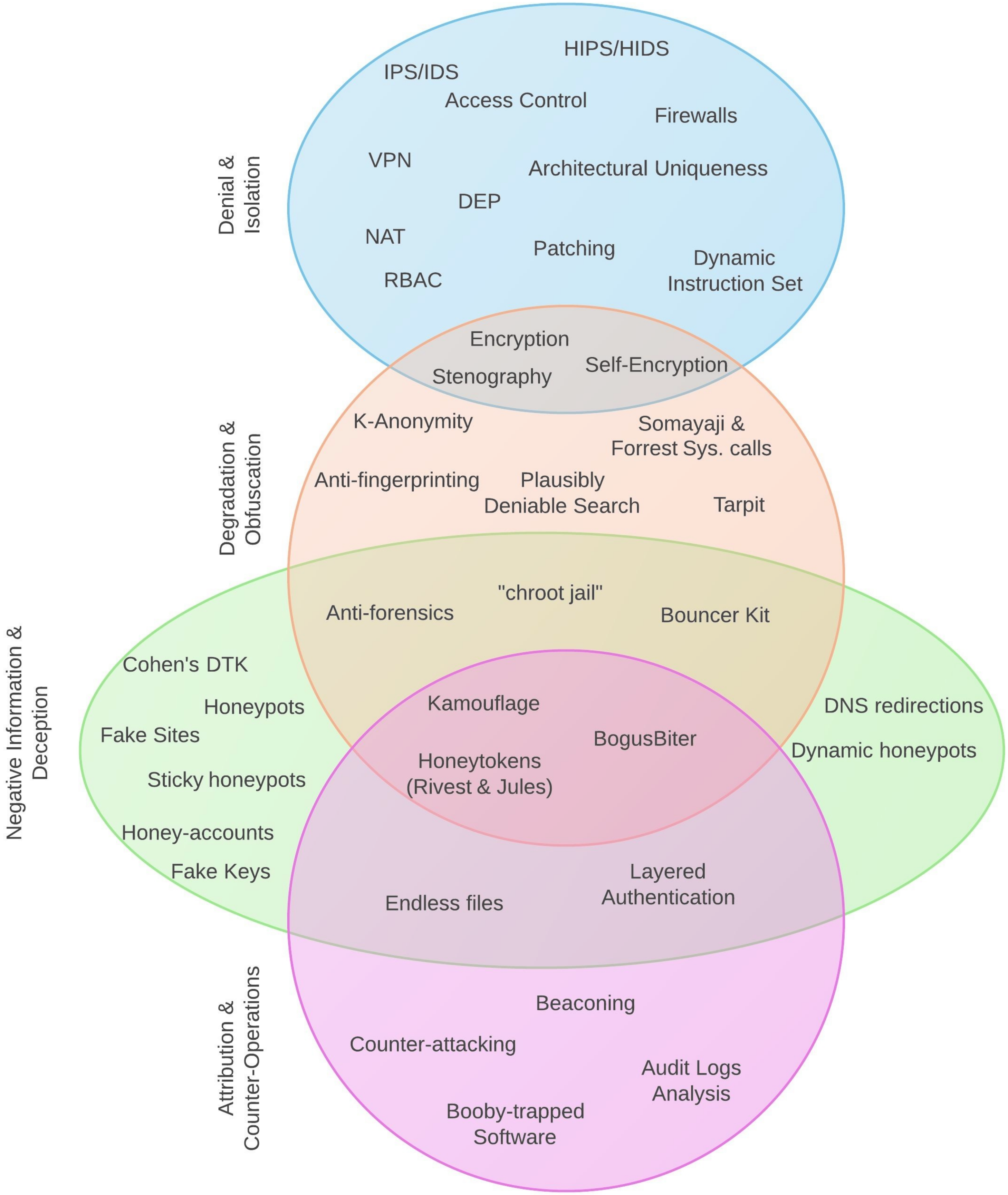
[redacted]

Deception in conflict is an old, tried-and-true concept!

The same is true in Cyber Security.

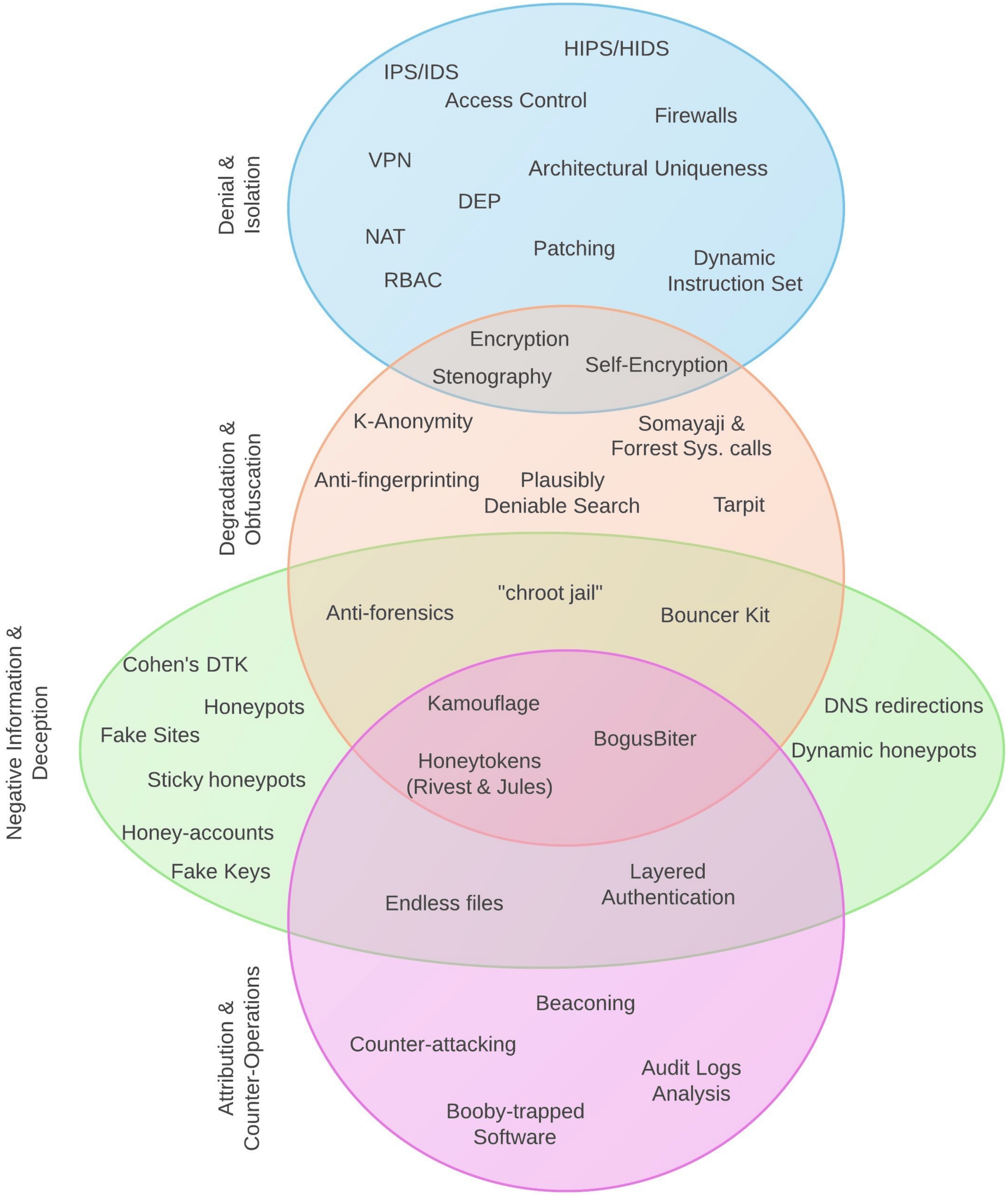
Using Deception

We have developed a classification scheme for describing how deception has been employed in cyber…and suggests where it might be added.



Using Deception

- **Denial/Isolation —>** Prevent and Hide



Denial & Isolation

IPS/IDS

HIPS/HIDS

Access Control

Firewalls

VPN

Architectural Uniqueness

DEP

NAT

Patching

RBAC

Dynamic Instruction Set

Encryption

Stenography

Self-Encryption

Degradation & Obfuscation

K-Anonymity

Somayaji & Forrest Sys. calls

Anti-fingerprinting

Plausibly Deniable Search

Tarpit

"chroot jail"

Anti-forensics

Bouncer Kit

Negative Information & Deception

Cohen's DTK

Honeypots

Fake Sites

Sticky honeypots

Honey-accounts

Fake Keys

DNS redirections

Dynamic honeypots

Kamouflage

BogusBiter

Honeytokens (Rivest & Jules)

Endless files

Layered Authentication

Attribution & Counter-Operations

Beaconing

Counter-attacking

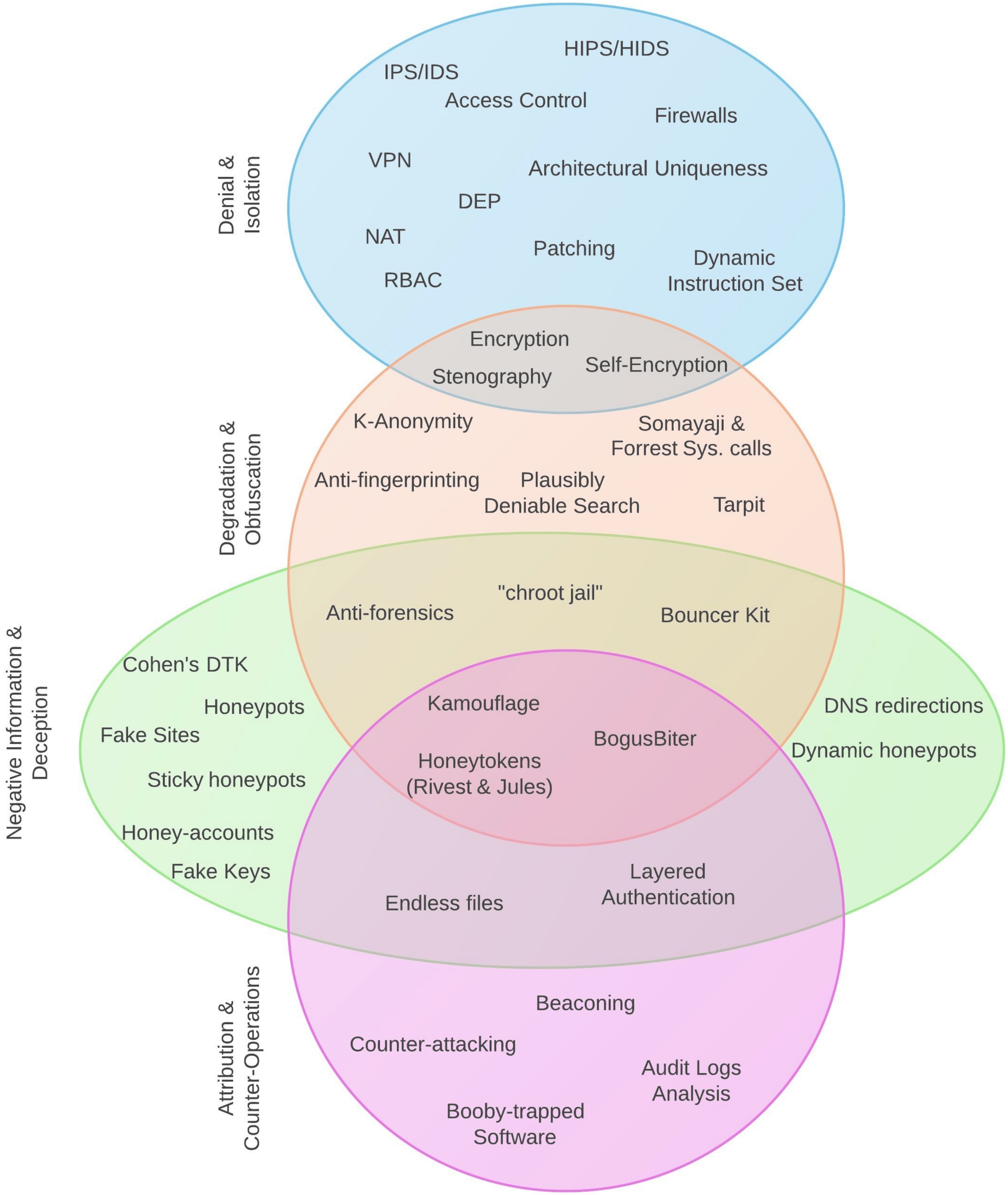
Audit Logs Analysis

Booby-trapped Software

Using Deception

- **Denial/Isolation —>** Prevent and Hide

- **Degradation/Obfuscation —>** Slow, Reduce Recovery, Obfuscate and Create Noise



Denial & Isolation

HIPS/HIDS
IPS/IDS
Access Control
Firewalls
VPN
Architectural Uniqueness
DEP
NAT
Patching
RBAC
Dynamic Instruction Set

Encryption
Stenography
Self-Encryption

Degradation & Obfuscation

K-Anonymity
Somayaji & Forrest Sys. calls
Anti-fingerprinting
Plausibly Deniable Search
Tarpit

"chroot jail"
Anti-forensics
Bouncer Kit

Negative Information & Deception

Cohen's DTK
Honeypots
Fake Sites
Sticky honeypots
Honey-accounts
Fake Keys
DNS redirections
Dynamic honeypots

Kamouflage
BogusBiter
Honeytokens (Rivest & Jules)

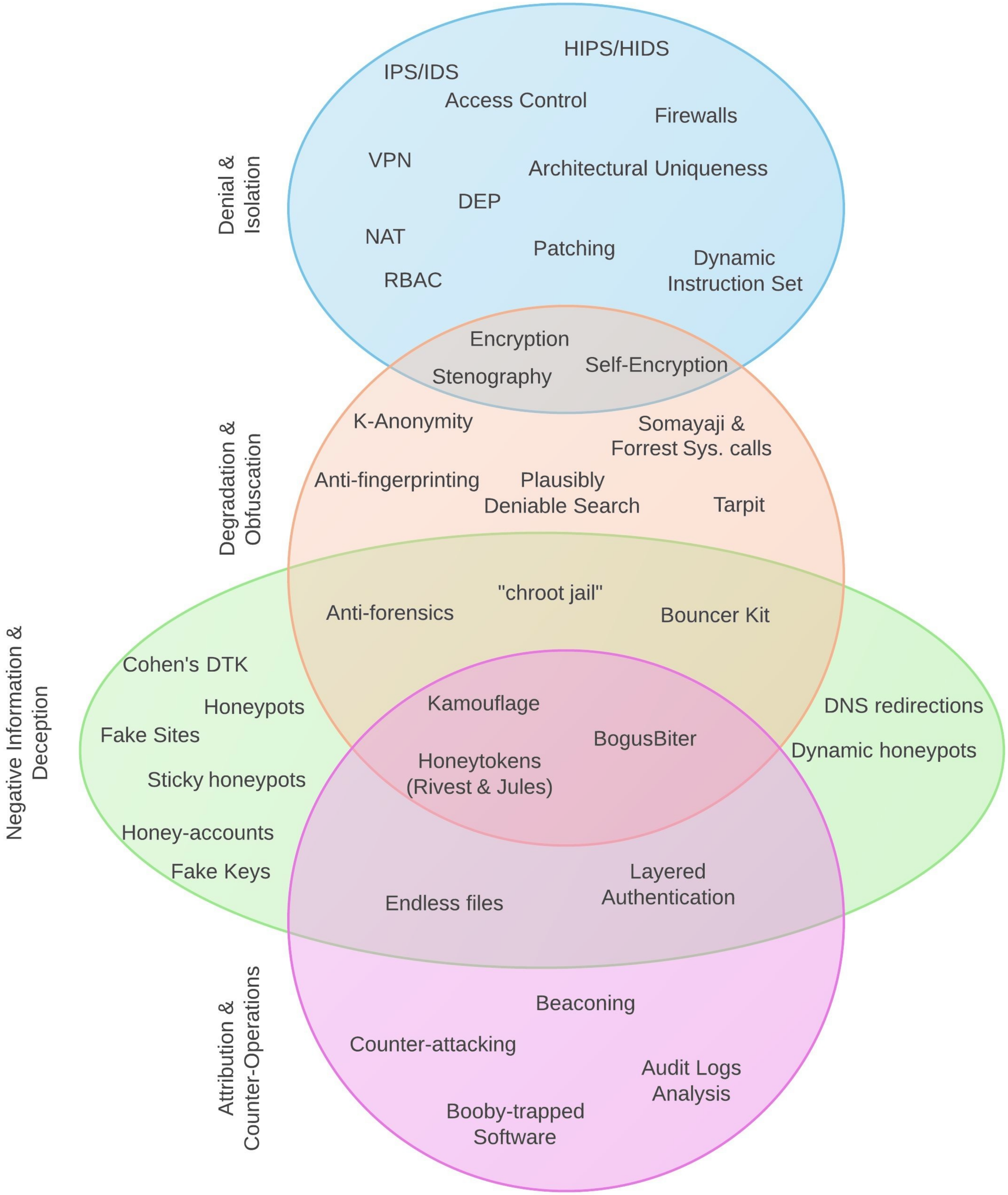
Endless files
Layered Authentication

Attribution & Counter-Operations

Beaconing
Counter-attacking
Audit Logs Analysis
Booby-trapped Software

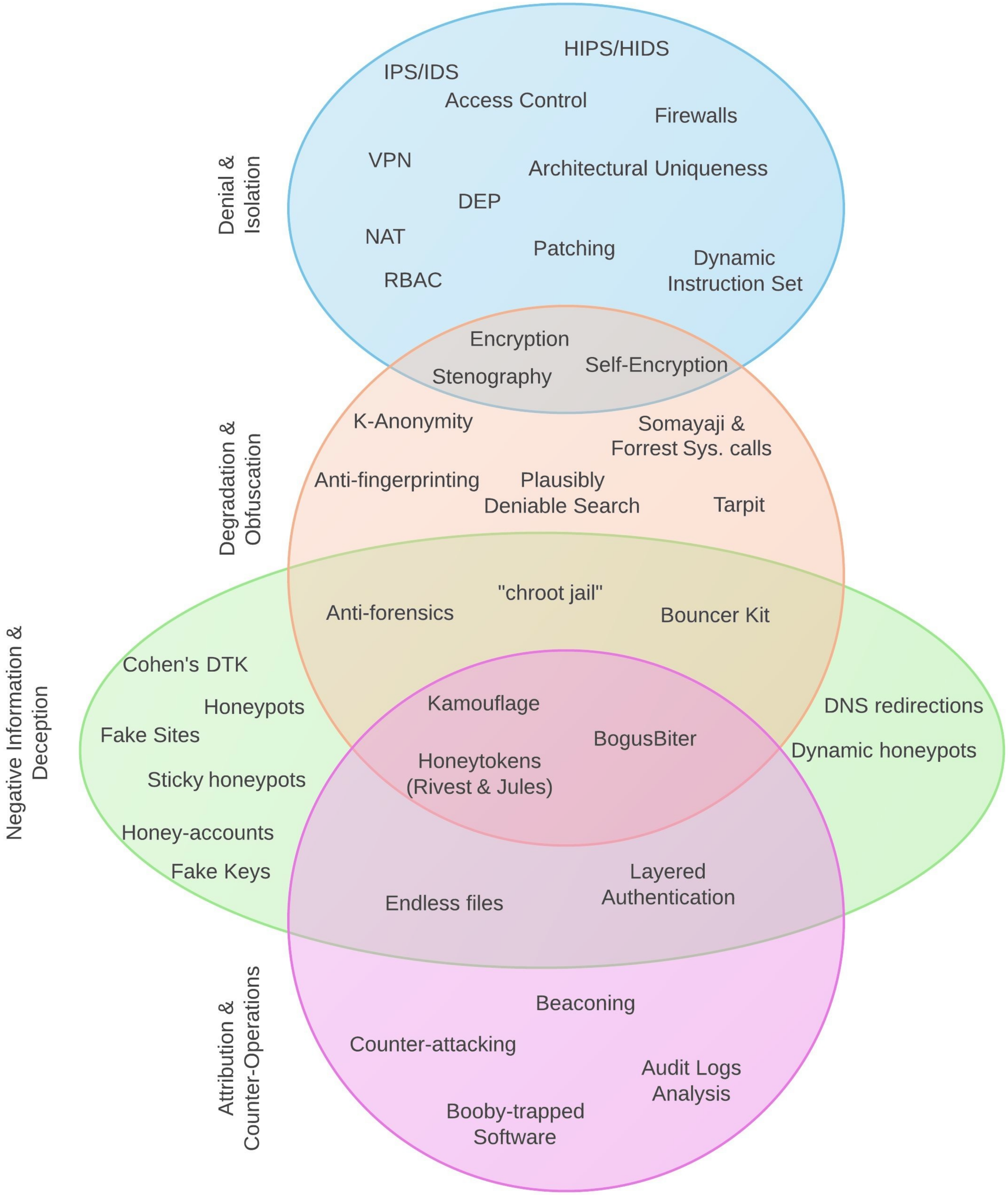
Using Deception

- **Denial/Isolation —>** Prevent and Hide
- **Degradation/Obfuscation —>** Slow, Reduce Recovery, Obfuscate and Create Noise
- **Deception/Negative Info. —>** Lead Astray, Decoy and Add Risk



Using Deception

- **Denial/Isolation —>** Prevent and Hide
- **Degradation/Obfuscation —>** Slow, Reduce Recovery, Obfuscate and Create Noise
- **Deception/Negative Info. —>** Lead Astray, Decoy and Add Risk
- **Attribution/Counter-Op. —>** Attribute and Cause Damage

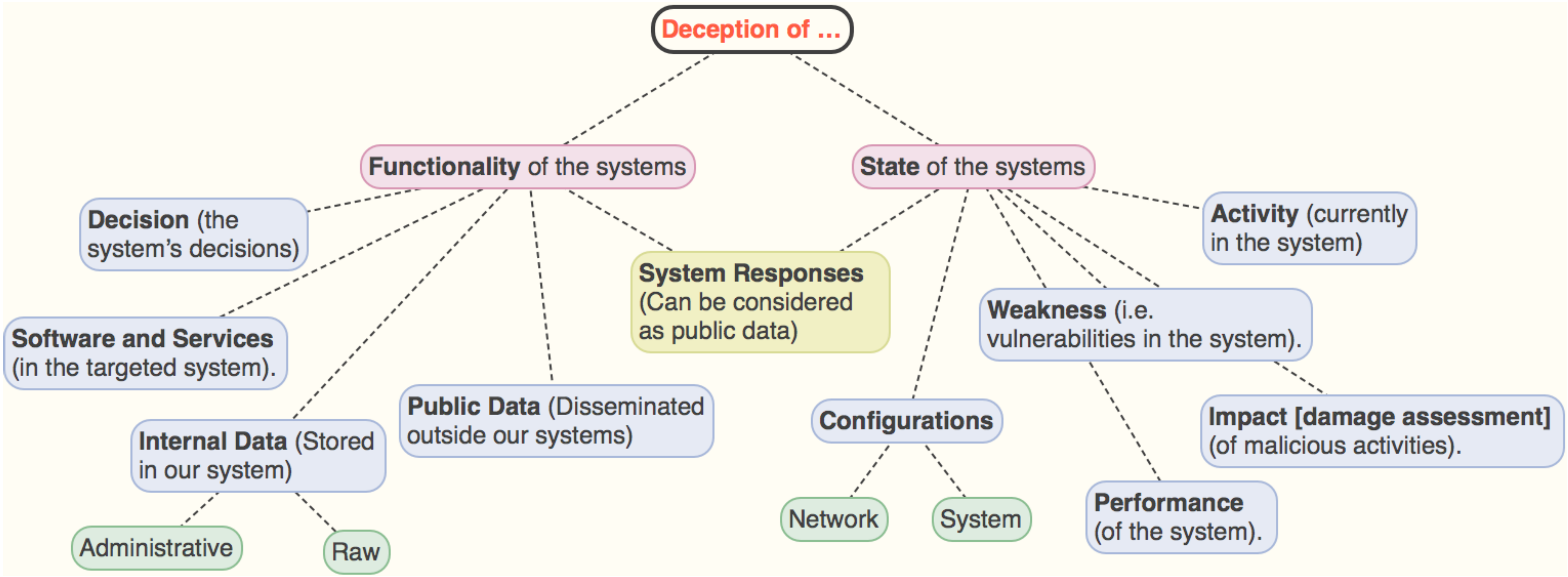


Deception to Improve Security

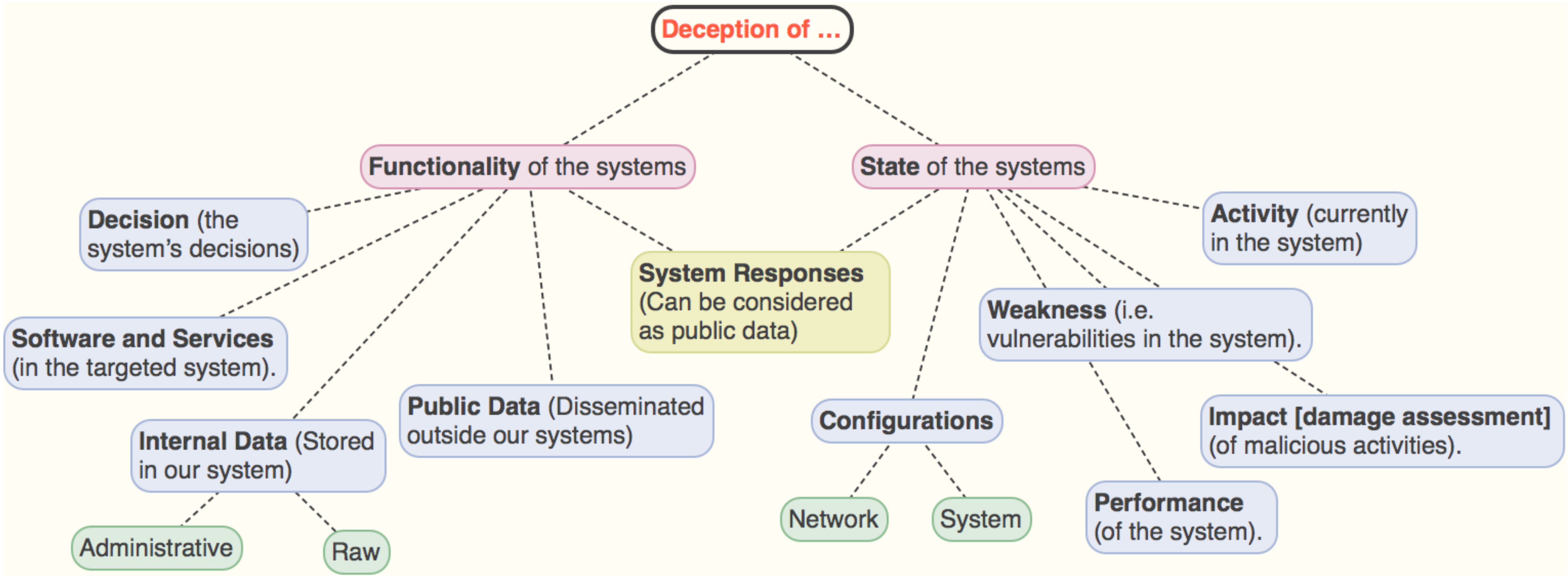
- Used as ad-hoc attempt:
  - Deception has been mainly used as “**trapping**” or “**deterrence**” tools.

- Traditional security (-) and deception (+) work in tandem.

- Three special advantages:
  1. Increase entropy when there is leakage.
  2. Gain information about adversaries.
  3. Gives defenders an edge in OODA.


Deception of …
Functionality of the systems
State of the systems
Decision (the system's decisions)
Software and Services (in the targeted system).
Internal Data (Stored in our system)
Public Data (Disseminated outside our systems)
System Responses (Can be considered as public data)
Administrative
Raw
Activity (currently in the system)
Weakness (i.e. vulnerabilities in the system).
Configurations
Network
System
Impact [damage assessment] (of malicious activities).
Performance (of the system).

Deception Model



Deception of …

Functionality of the systems

Decision (the system's decisions)

Software and Services (in the targeted system).

Internal Data (Stored in our system)

Administrative

Raw

Public Data (Disseminated outside our systems)

System Responses (Can be considered as public data)

State of the systems

Configurations

Network

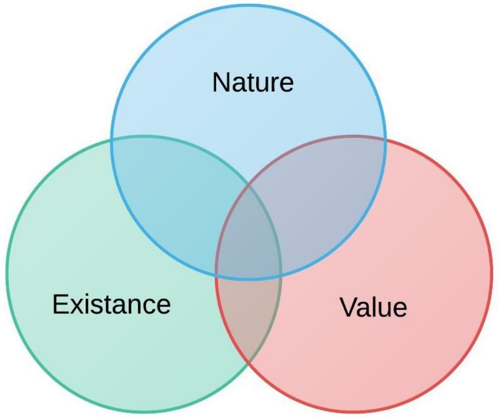
System

Weakness (i.e. vulnerabilities in the system).

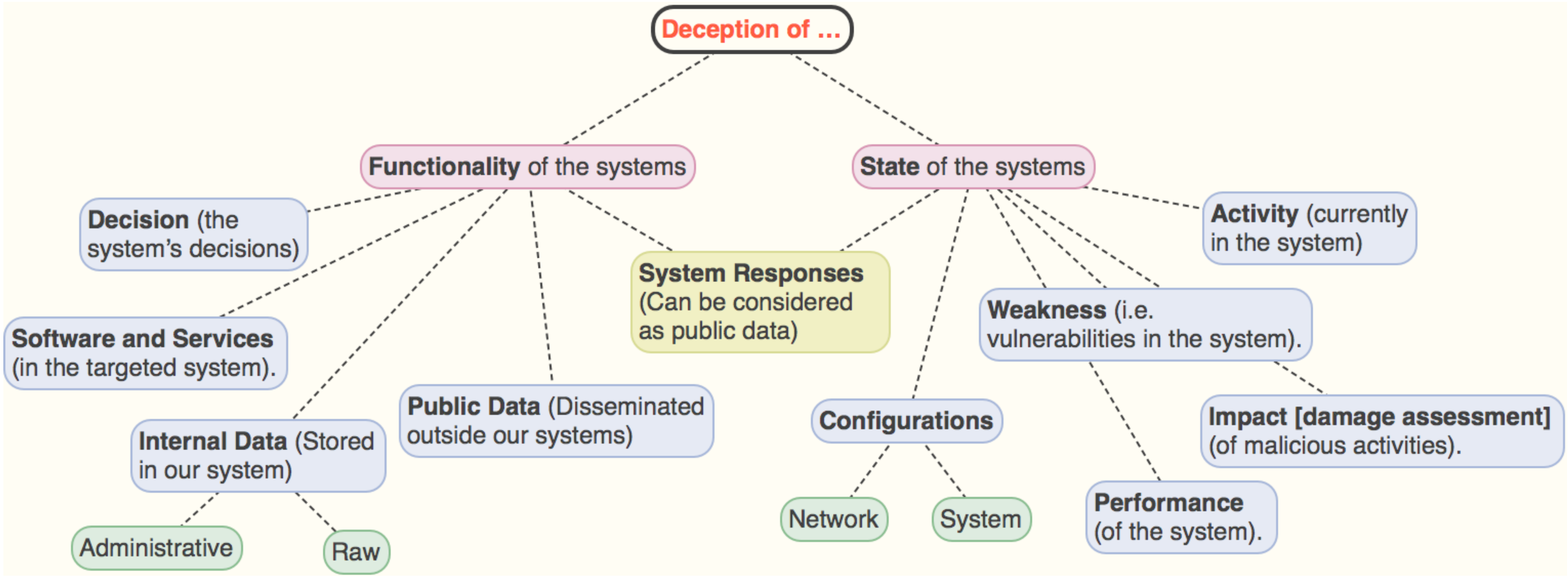
Performance (of the system).

Impact [damage assessment] (of malicious activities).

Activity (currently in the system)



Deception Model



Deception of …

Functionality of the systems

State of the systems

Decision (the system's decisions)

Software and Services (in the targeted system).

Internal Data (Stored in our system)

Administrative

Raw

Public Data (Disseminated outside our systems)

System Responses (Can be considered as public data)

Configurations

Network

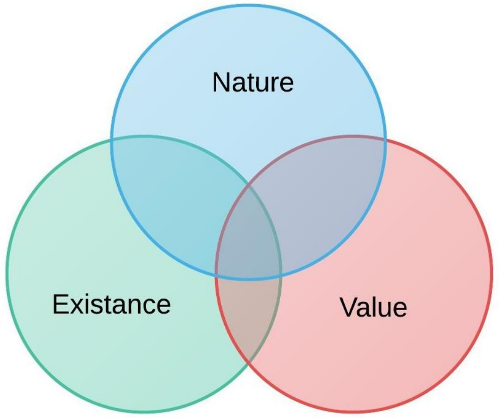
System

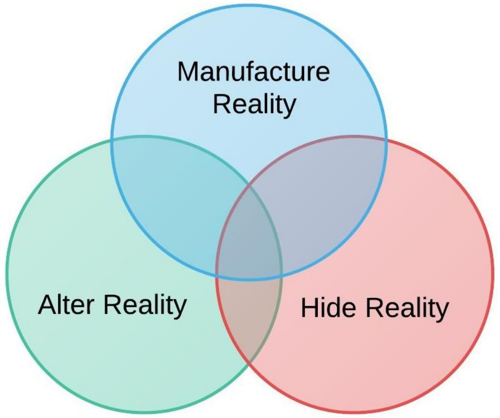
Weakness (i.e. vulnerabilities in the system).

Performance (of the system).

Activity (currently in the system)

Impact [damage assessment] (of malicious activities).





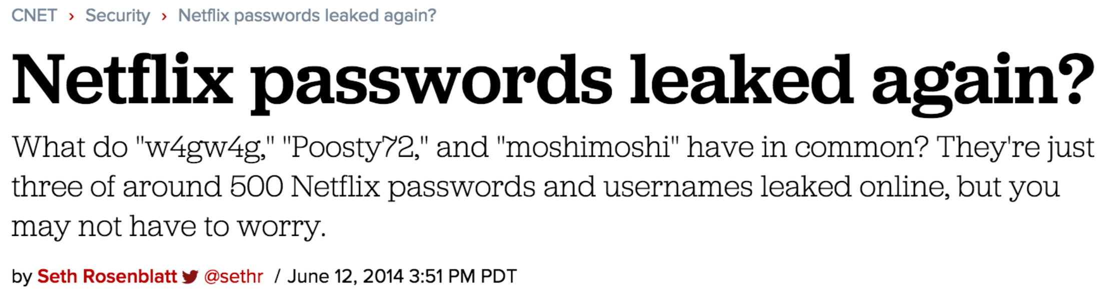
Deception Model

Some of Our Current Projects

Ersatz Passwords — Ending Password Cracking

- Passwords files are stolen and leaked all the time

- Can we make a password file return fake passwords when cracked and detect that upon login?

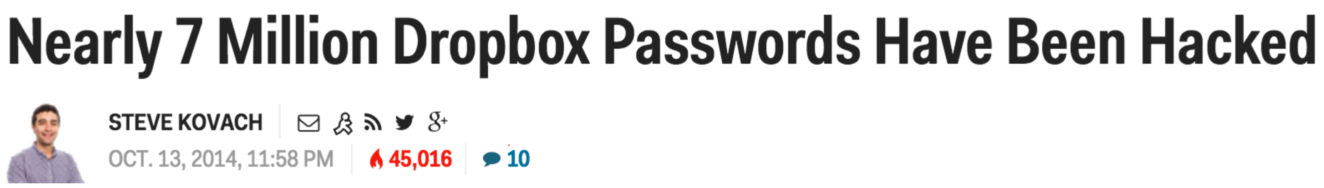


CNET › Security › Netflix passwords leaked again?

Netflix passwords leaked again?

What do "w4gw4g," "Poosty72," and "moshimoshi" have in common? They're just three of around 500 Netflix passwords and usernames leaked online, but you may not have to worry.

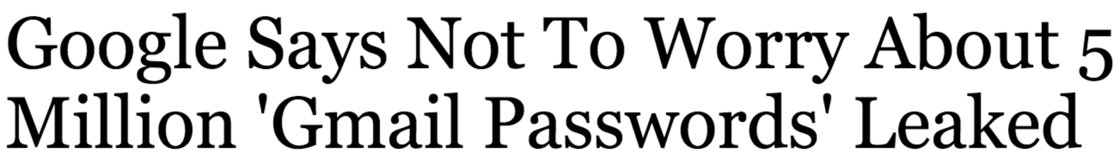
by Seth Rosenblatt @sethr / June 12, 2014 3:51 PM PDT



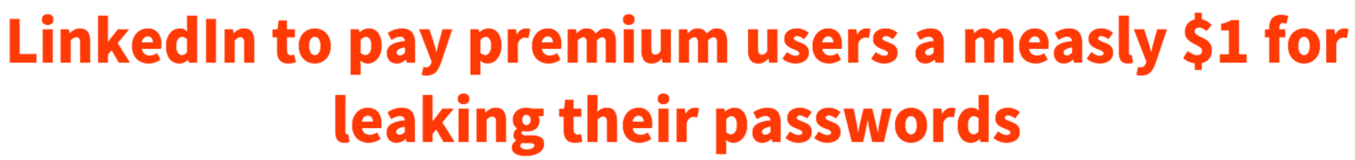
Nearly 7 Million Dropbox Passwords Have Been Hacked

STEVE KOVACH

OCT. 13, 2014, 11:58 PM | 45,016 | 10



Google Says Not To Worry About 5 Million 'Gmail Passwords' Leaked

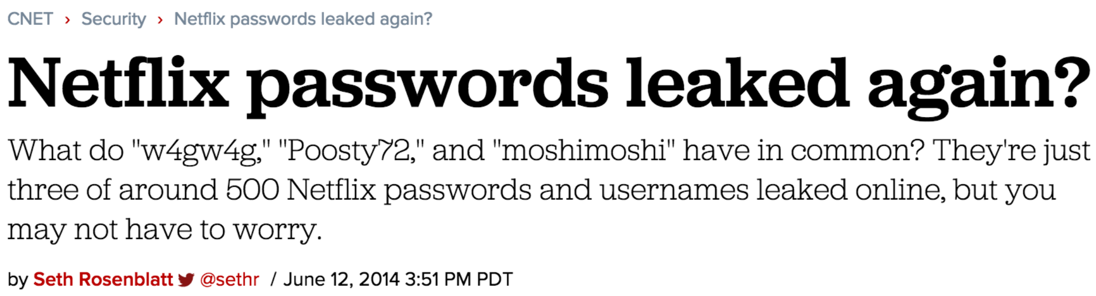


LinkedIn to pay premium users a measly $1 for leaking their passwords

Ersatz Passwords — Ending Password Cracking

- Passwords files are stolen and leaked all the time
- Can we make a password file return fake passwords when cracked and detect that upon login?

Yes we can!

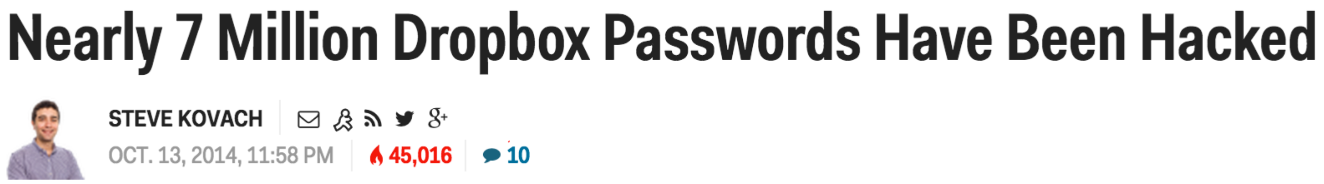


CNET › Security › Netflix passwords leaked again?

Netflix passwords leaked again?

What do "w4gw4g," "Poosty72," and "moshimoshi" have in common? They're just three of around 500 Netflix passwords and usernames leaked online, but you may not have to worry.

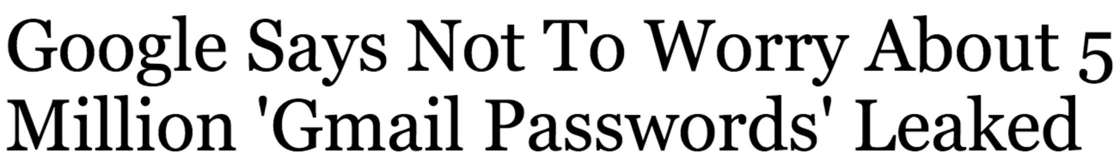
by Seth Rosenblatt @sethr / June 12, 2014 3:51 PM PDT



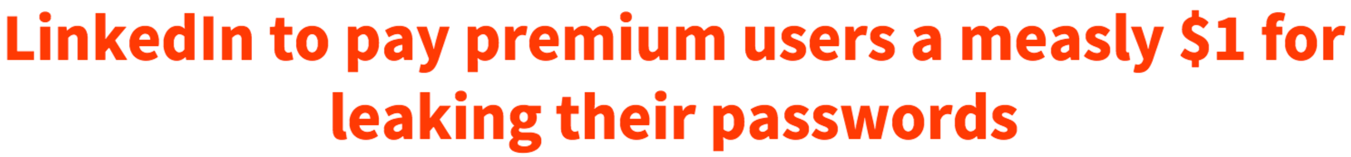
Nearly 7 Million Dropbox Passwords Have Been Hacked

STEVE KOVACH

OCT. 13, 2014, 11:58 PM 45,016 10

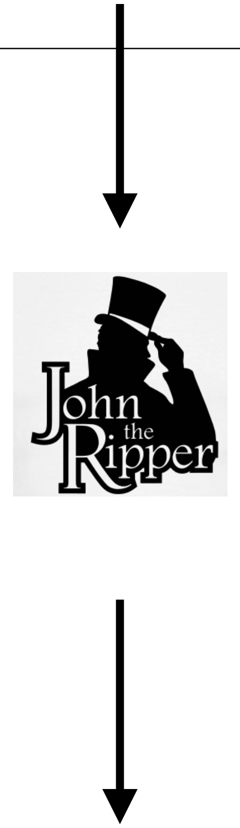


Google Says Not To Worry About 5 Million 'Gmail Passwords' Leaked



LinkedIn to pay premium users a measly $1 for leaking their passwords

```
…
root:$1$hnH/w50a$tPdv5HZRsDP46FtsW8eXH/:0:0::0:0::/root:/bin/csh
spaf:$1$7hstg1PAq$wTnskj1HwLgdD90SerkQp:0:0::0:0::/homes/spaf:/bin/sh
…
```

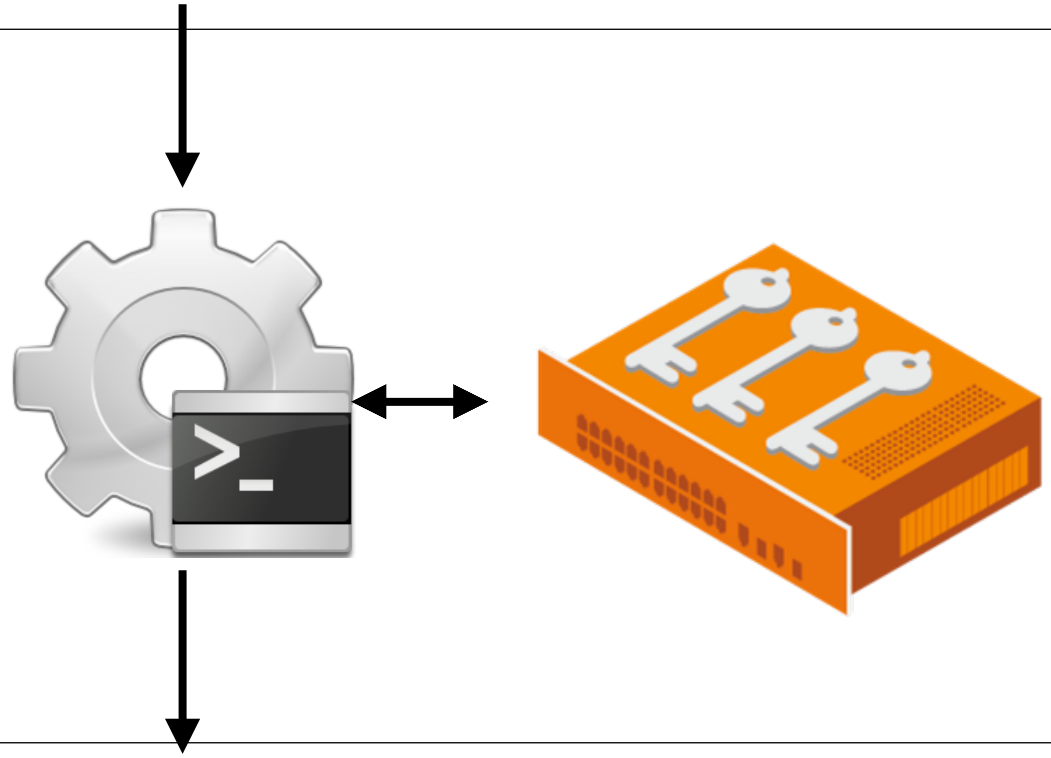


```
…
root w)oi2djl;Ksju
spaf $tR0ngP@s@w0rD
…
```

Ersatz Passwords

Current method

```
…
root:$1$hnH/w50a$tPdv5HZRsDP46FtsW8eXH/:0:0::0:0::/root:/bin/csh
spaf:$1$7hstg1PAq$wTnskj1HwLgdD90SerkQp:0:0::0:0::/homes/spaf:/bin/sh
…
```



```
…
root:$1$Afeo2MkL$tWoL9yeQabg2luyJhRWlp:0:0::0:0::/root:/bin/csh
spaf:$1$9LksuHq9$oKjhyD65SajuWGy68udGfo:0:0::0:0::/homes/spaf:/bin/sh
…
```

Ersatz Passwords

A practical Example

```
…
root:$1$Afeo2MkL$tWoL9yeQabg2luyJhRWlp:0:0::0:0::/root:/bin/csh
spaf:$1$9LksuHq9$oKjhyD65SajuWGy68udGfo:0:0::0:0::/homes/spaf:/bin/sh
…
```

```
…
root adsk(soa97Sd;
spaf W3@kPaWn:-)
…
```

Ersatz Passwords

With our method

In Slightly More Detail

1. Encrypt the real password with an HDF ß=HDF(pass)

2. Generate a fake password, p*

3.Generate new salt = ß ⊕ p* (so, ß ⊕ salt == p*)

4. Store ∂ = H(p* || salt),salt in password file

5. Enter p’
   (a) If H(p’ || salt) == ∂ **Alarm!**
   (b) If HDF(p’) ⊕ salt == p*   **OK!**
   (c) Otherwise, “**Bad ID or Password**”

Patches

- Patches are made to software
  - Security
  - Performance
  - Bug fixes
- Remove the issue/bug
- Can we use patches to our advantage?

Patches

- Patches are made to software
  - Security
  - Performance
  - Bug fixes
- Remove the issue/bug
- Can we use patches to our advantage?

Yes we can!

Deceptive Patches

- Introduce patches that respond deceptively
  - Fix the issue at hand as well as add an extra layer of security
  - Predict the adversary’s actions and respond accordingly

Deceptive Patches

- Benefits
  - Protect confidential data indirectly
  - Predict and monitor adversary’s movements
  - Prosecute attackers
  - Works against insiders, too
- Cons
  - Difficult to attain consistency

Deceptive patch example

```
…
n = dn_expand(msg, eom, cp+18, (char *)cp1, (sizeof data) - 18);

/* n is the length of the compressed domain name as seen in msg*/

printf("dn_expand returned:  %d, expanded name = %s\n", n, (char *)cp1);

if (n < 0) {
      printf("ERROR: n = %d < 0!\n", n);
      printf("EXITING RREXTRACT!\n");
      hp->rcode = FORMERR;
      return (-1);
}
…
n = dlen - (NS_SIG_SIGNER + n);
…
memcpy(cp1, cp, n);
```

Assignment of some calculated value to n

Use the value of n without checking it in memcpy

http://samate.nist.gov/SARD/view_testcase.php?tID=1291

Normal Patch

```
...
n = dlen - (NS_SIG_SIGNER + n);
...

if (n < 0){
    printf("ERROR: n = %d < 0!\n", n);
    printf("EXITING RREXTRACT!\n");
    hp->rcode = FORMERR;
    return (-1);
}

memcpy(cp1, cp, n);
```

Deceptive Patch

```
…
n = dlen - (NS_SIG_SIGNER + n);
…

if (n < 0){
// Respond deceptively by allowing the memcpy
to occur in a sandbox or present seg fault data
dump that is deceptive
// Exit current execution path
}

memcpy(cp1, cp, n);
```

Deception in Anti-Forensics

- Attackers rely on anti-forensics tools to remain hidden within a system
  - Example: Data purging
  - Example: Ephemeral tool placement

  Can we use deception to aid in forensics?

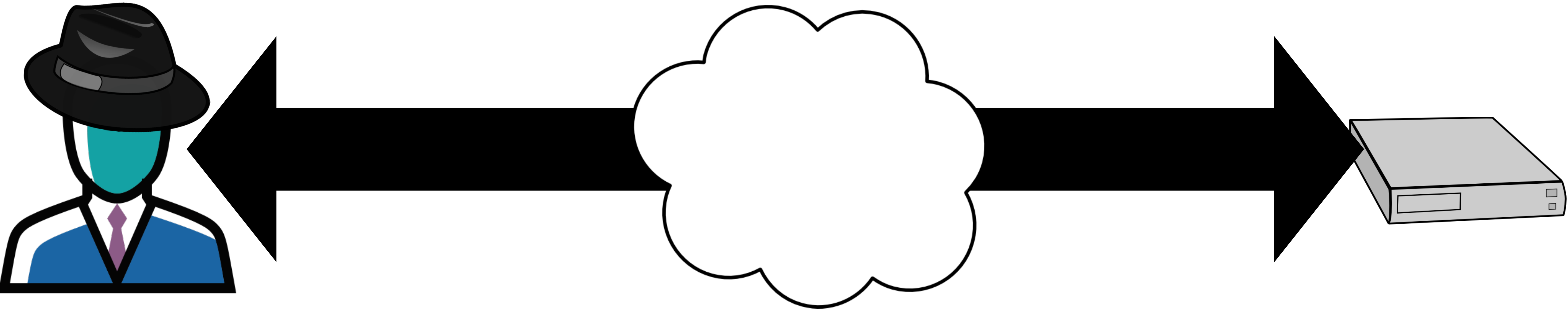
Deception in Anti-Forensics

- Attackers rely on anti-forensics tools to remain hidden within a system
  - Example: Data purging
  - Example: Ephemeral tool placement

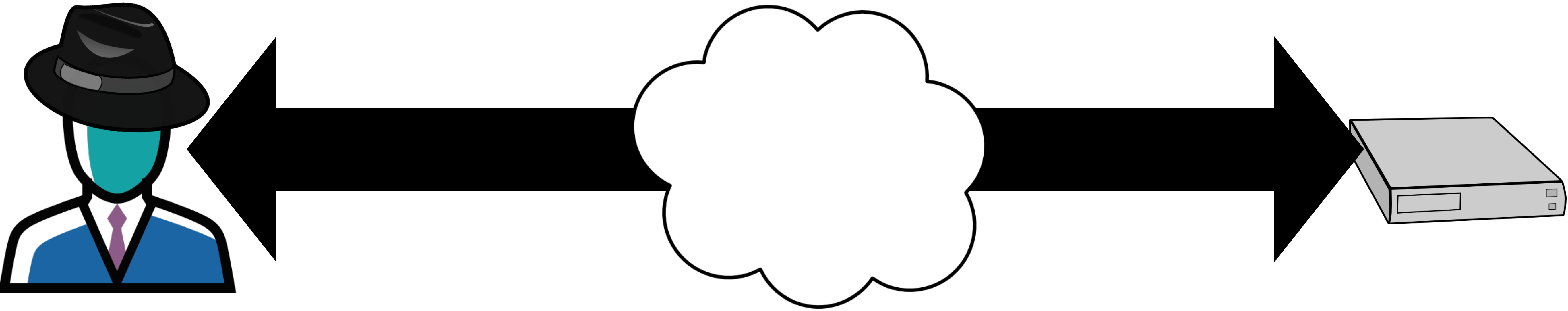
Can we use deception to aid in forensics?

Yes we can!

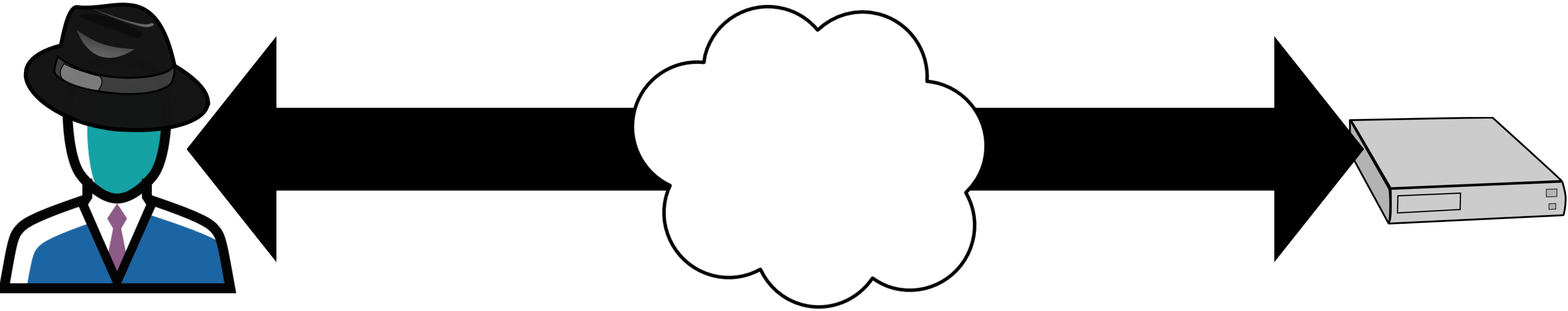
Deceptive Memory Systems



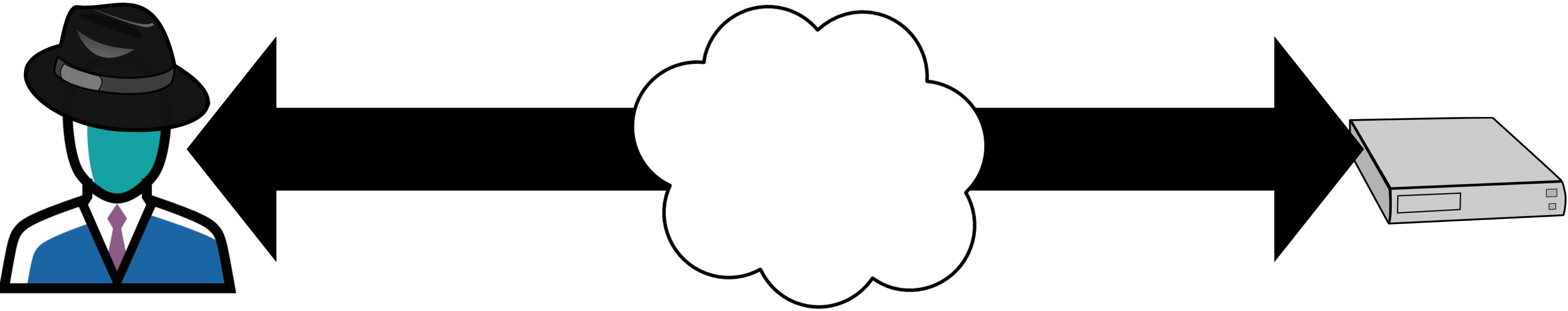
Deceptive Memory Systems



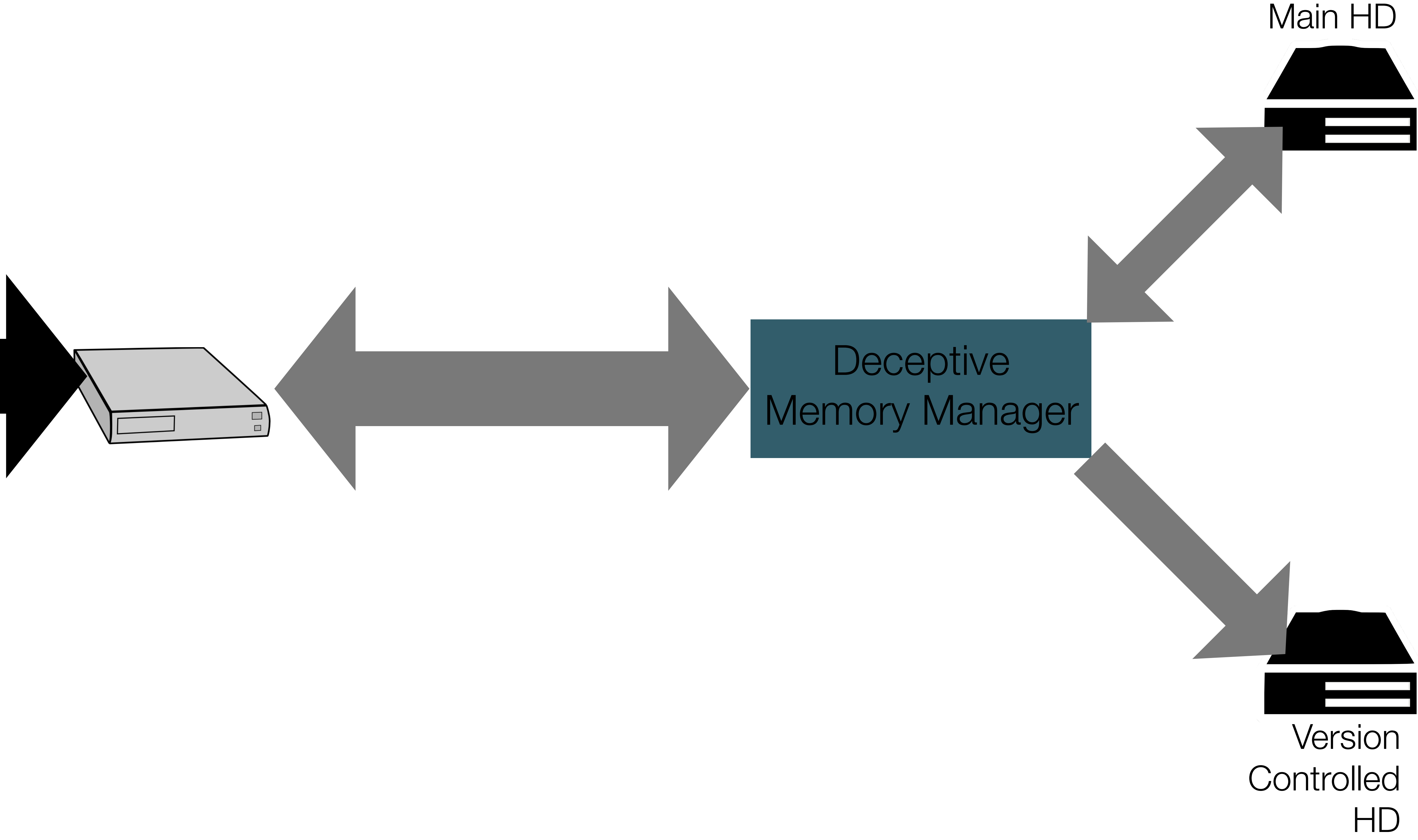
Deceptive Memory Systems



Deceptive Memory Systems



Deceptive Memory Systems

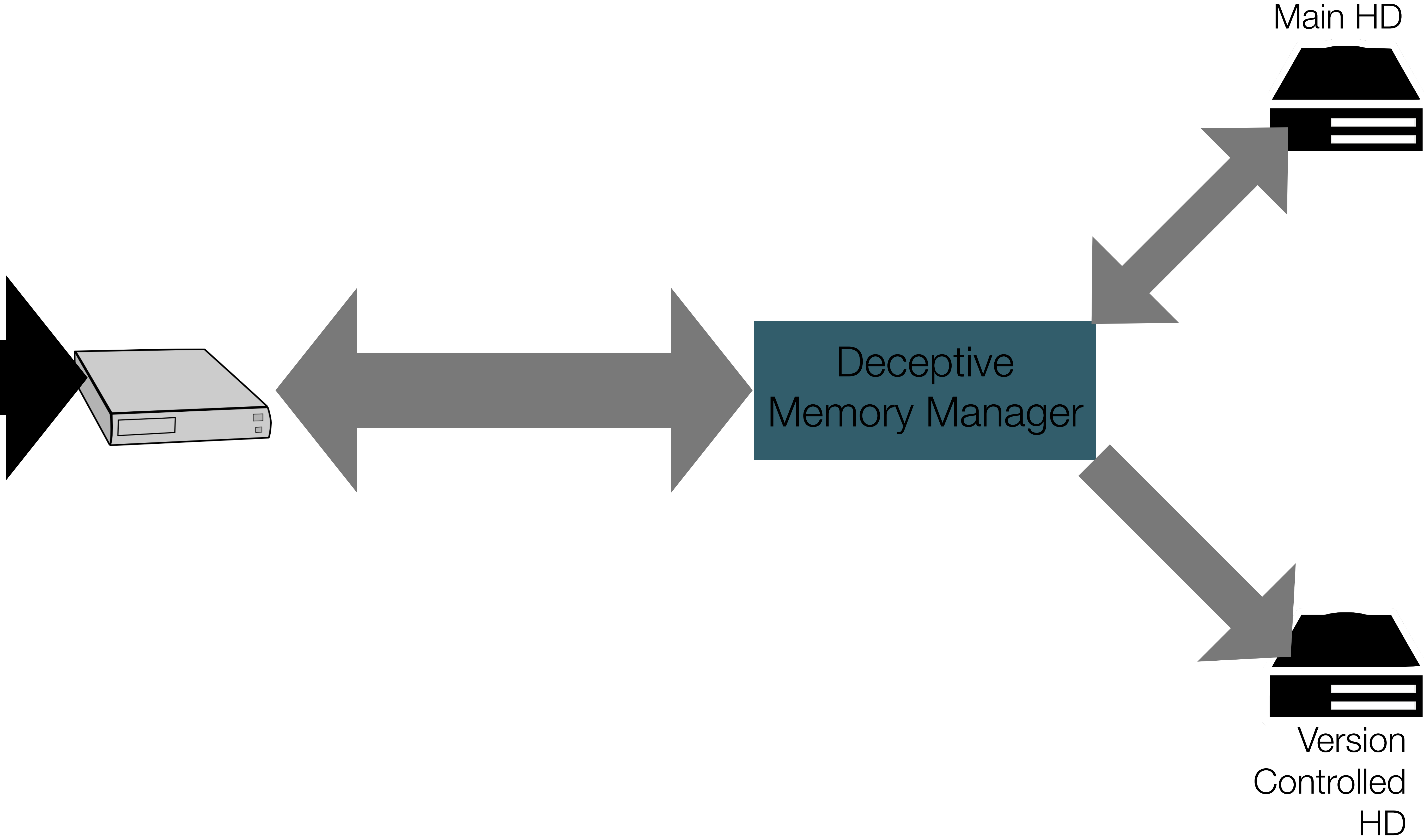


Main HD

Deceptive Memory Manager

Version Controlled HD

Deceptive Memory Systems

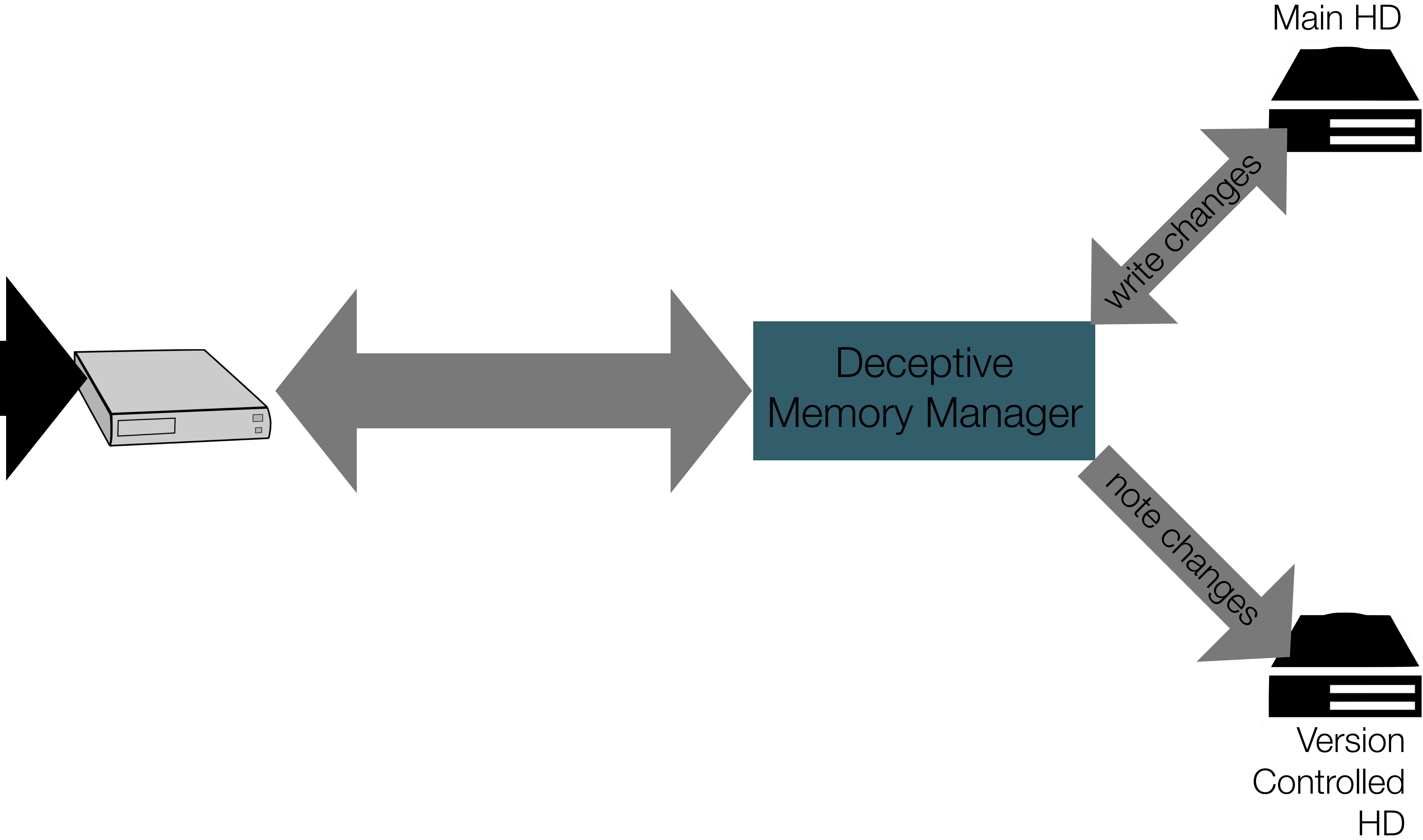


Main HD

Deceptive Memory Manager

Version Controlled HD

Deceptive Memory Systems

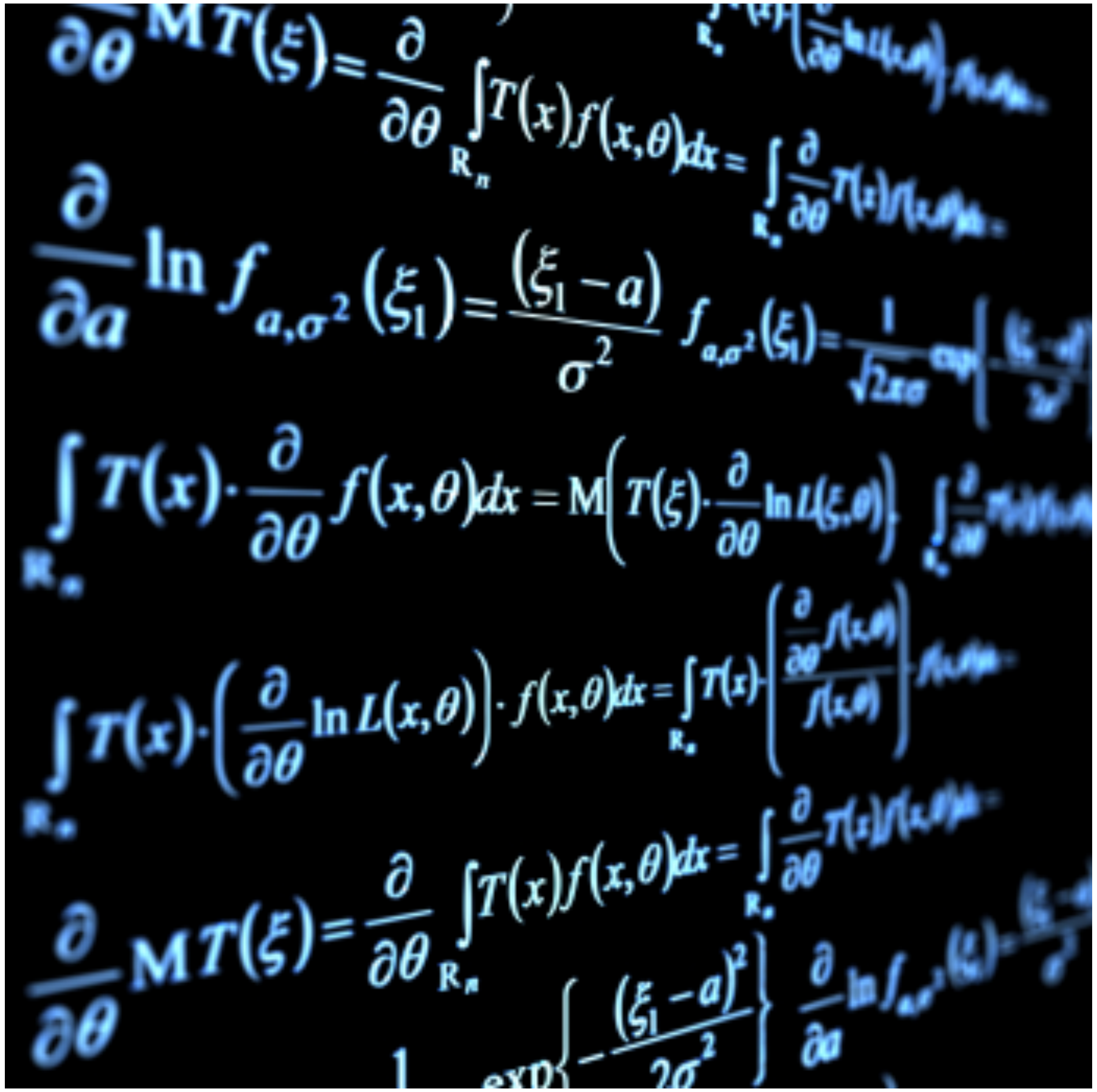


Deceptive Memory System - Challenges

- Identify behaviors of interest
- Maintain a minimum impact on performance
- Isolate the version control tracking from the attacker

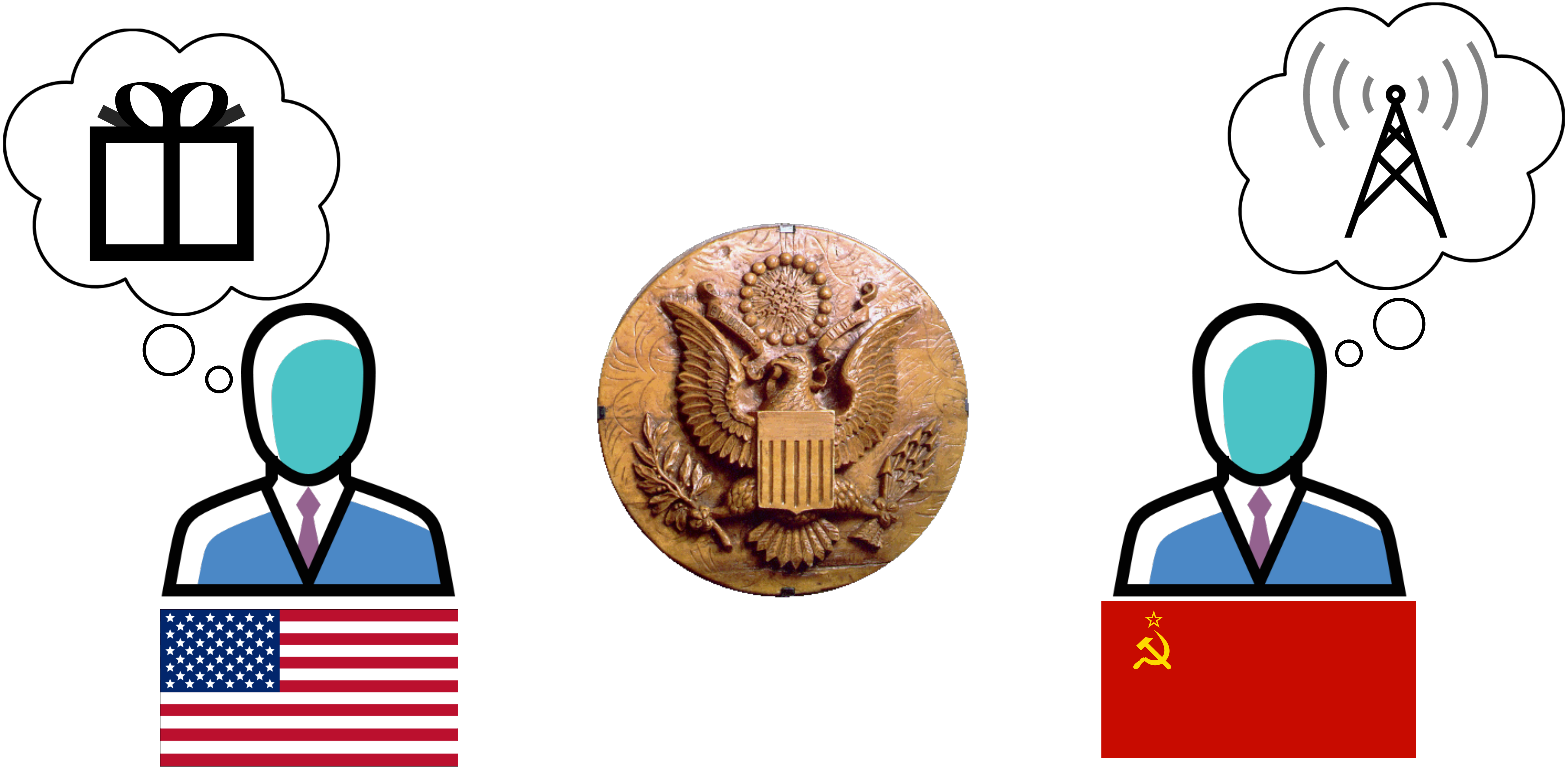
Modeling Deception in Information Security

- Analysis of conflicts where players have misconceptions of systems, assets, and intention of other players
- Analyze deception strategies to determine optimal defense
- Historically applied to military conflicts such as the Cuban Missile Crisis, Normandy Invasion, etc.

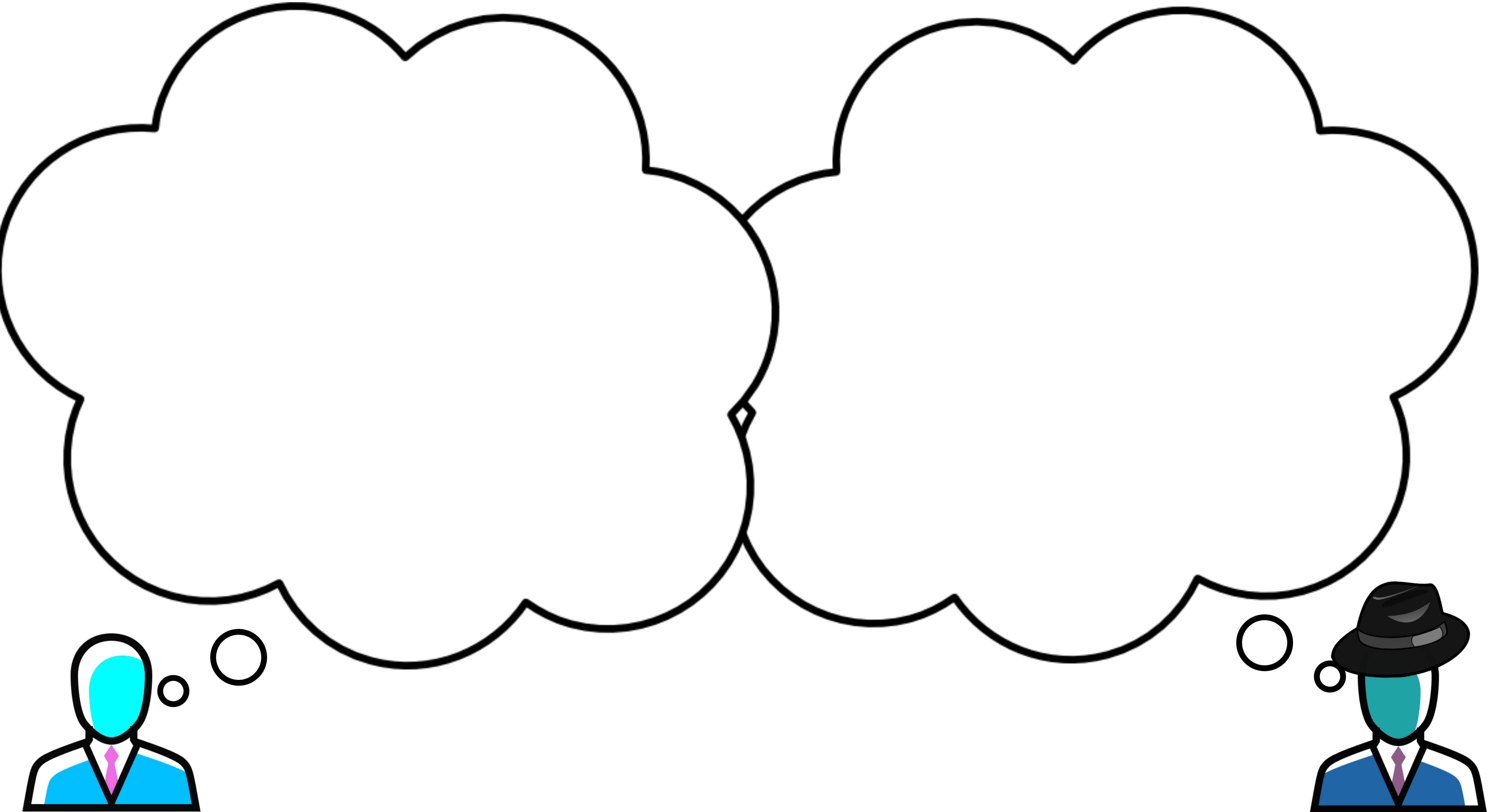


Hypergames

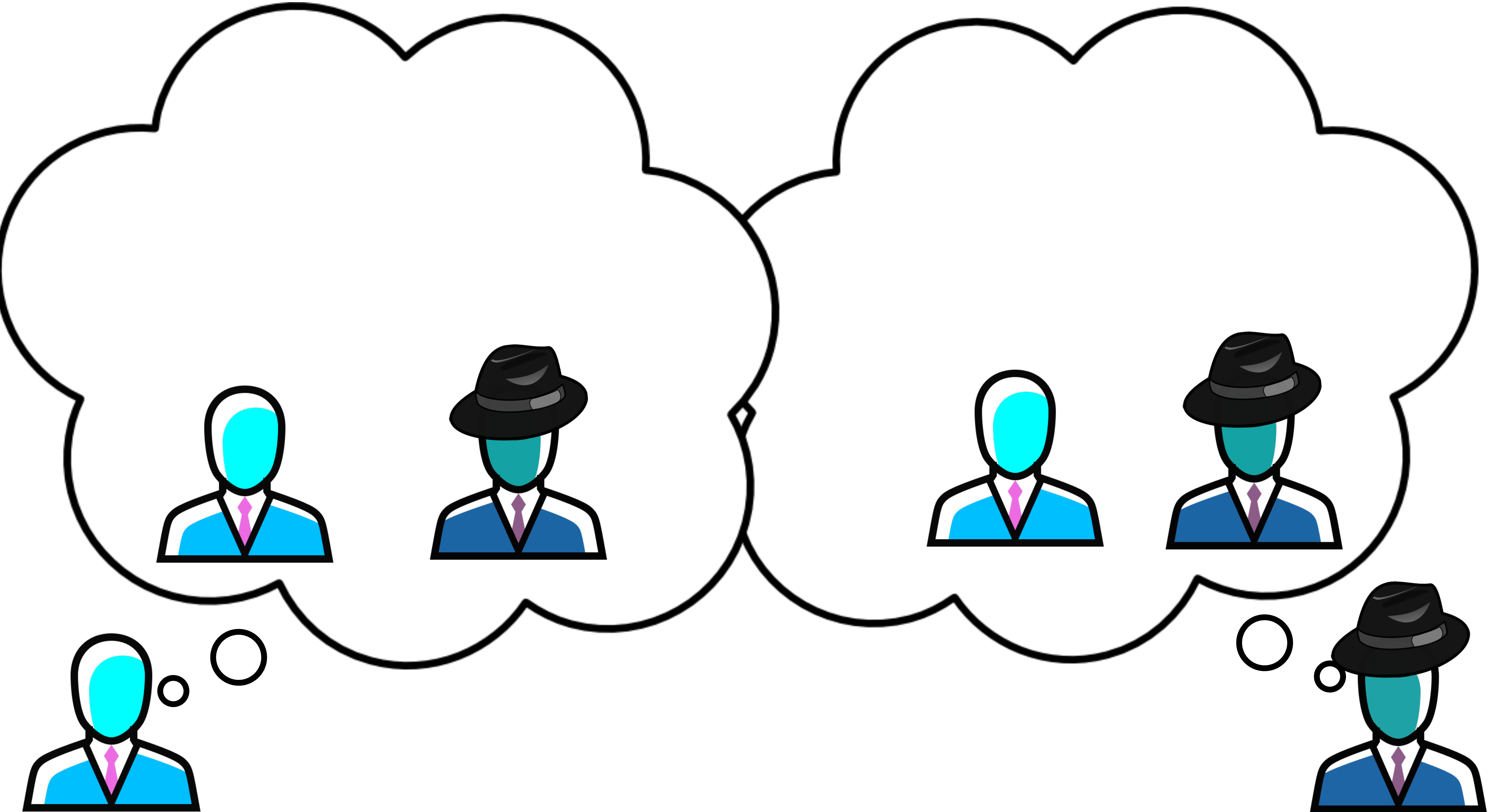
A game theoretic model where players may not understand the conflict



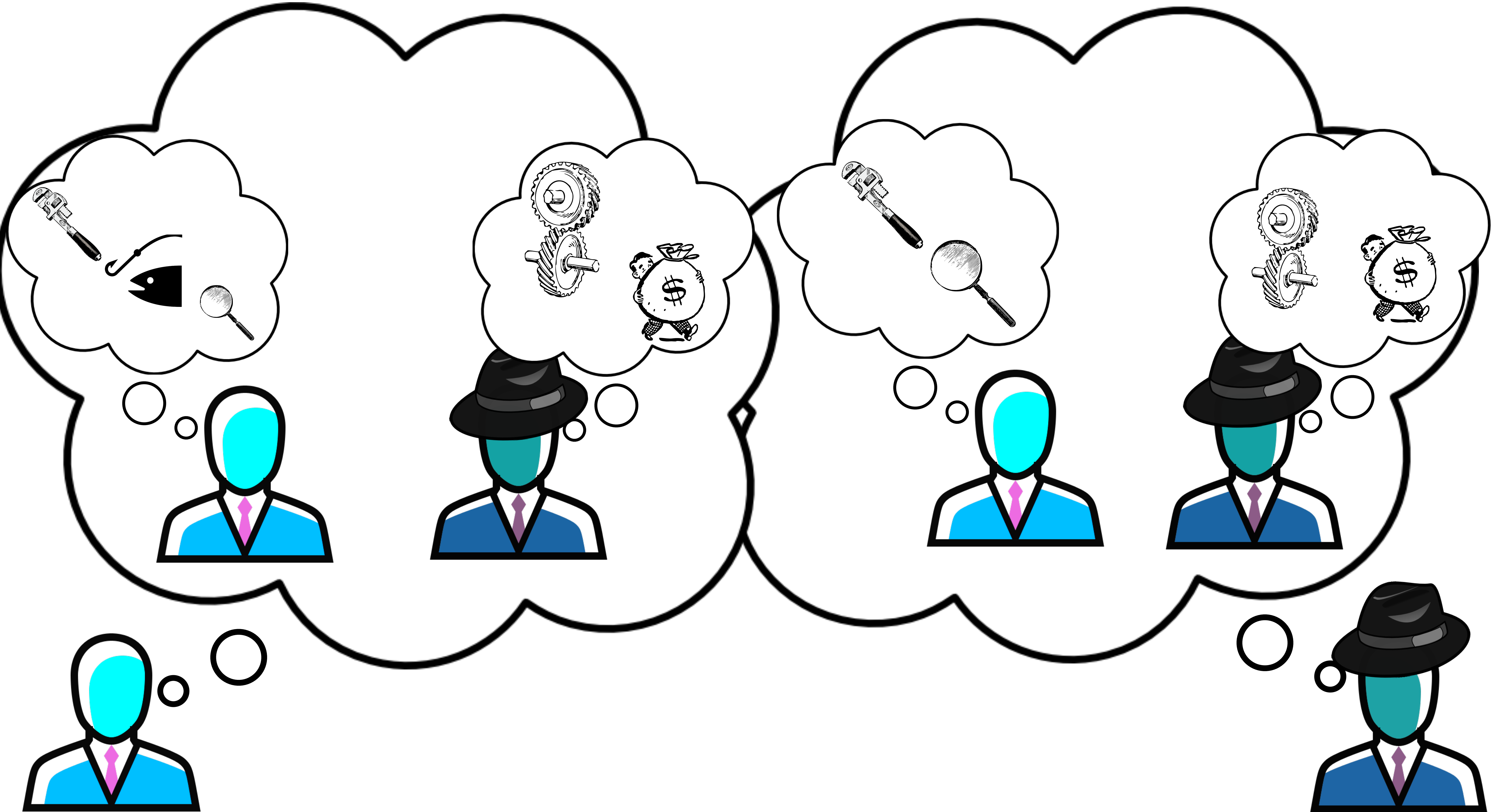
Hypergames - Equilibrium



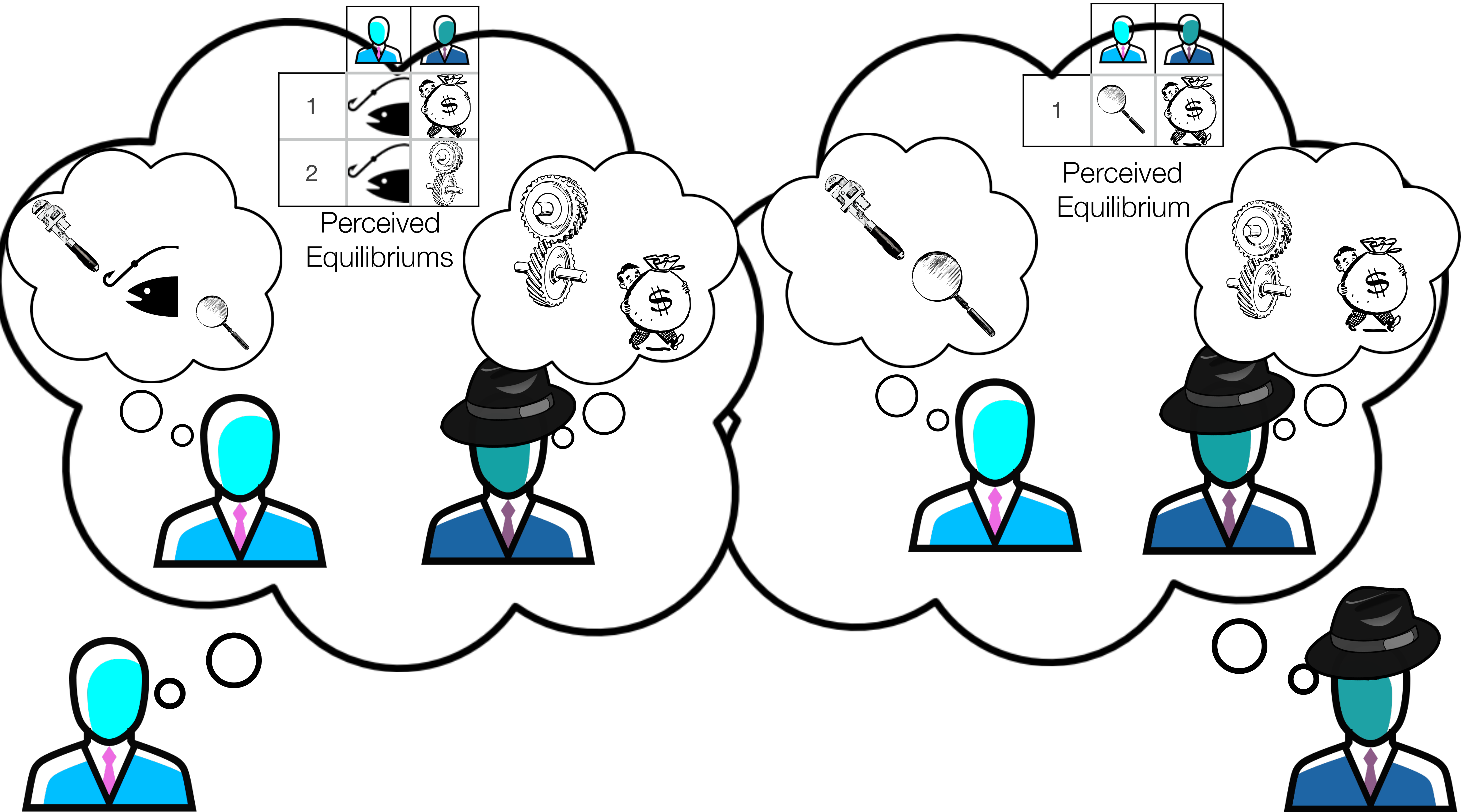
Hypergames - Equilibrium



Hypergames - Equilibrium



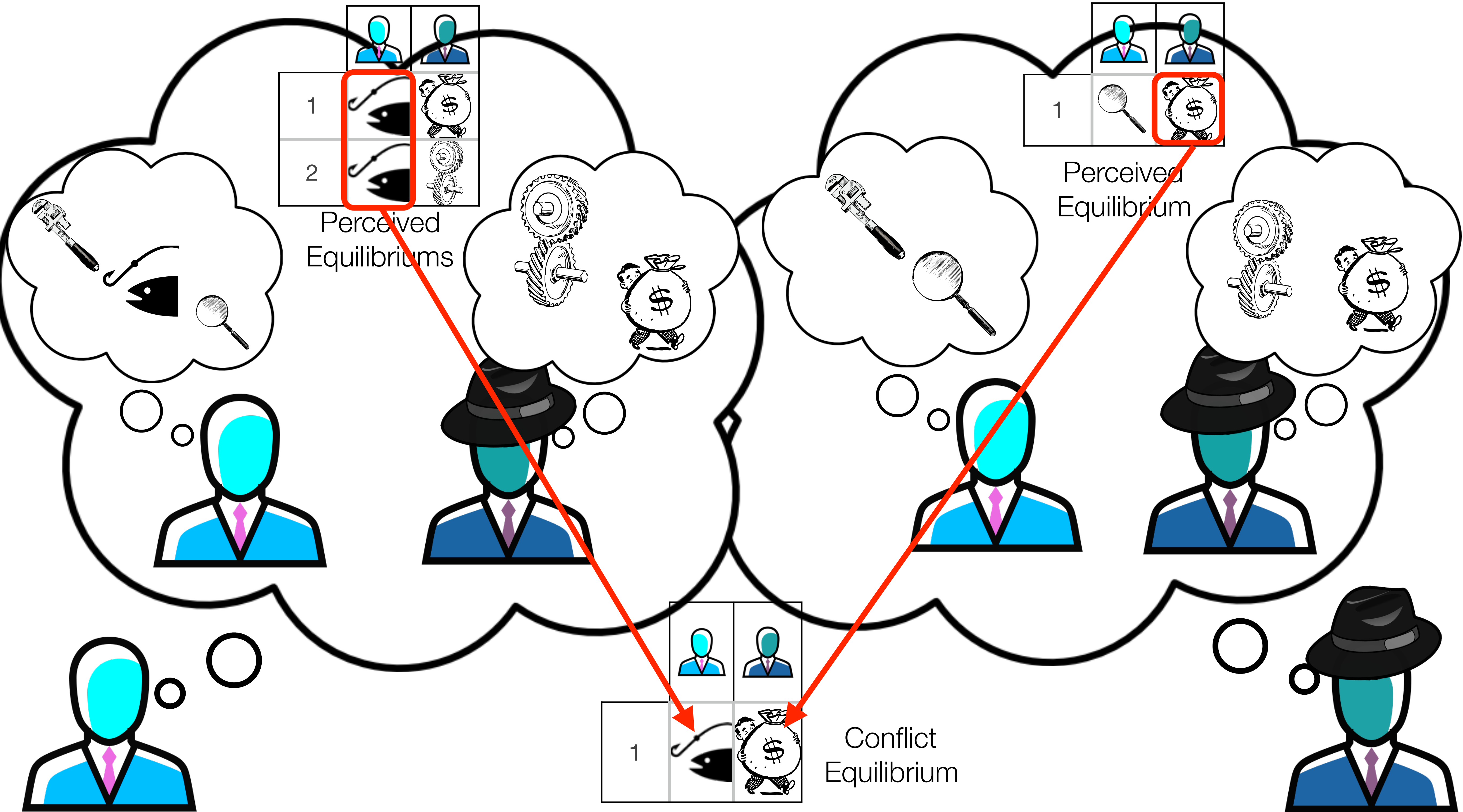
Hypergames - Equilibrium



Perceived Equilibriums

Perceived Equilibrium

Hypergames - Equilibrium



Hypergames Goals

- Design a flexible tool applicable to information security
- Analyze deceptive components in a defensive system
- Provide insight on level of effort needed to successfully deploy deception

There is potential for greater “security through obscurity.”

Questions? Comments? https://ceri.as/deception

There is potential for greater “security through obscurity.”

Trust us on that.
😈

Questions? Comments? https://ceri.as/deception