Tor Hidden Services

How Hidden is 'Hidden'?

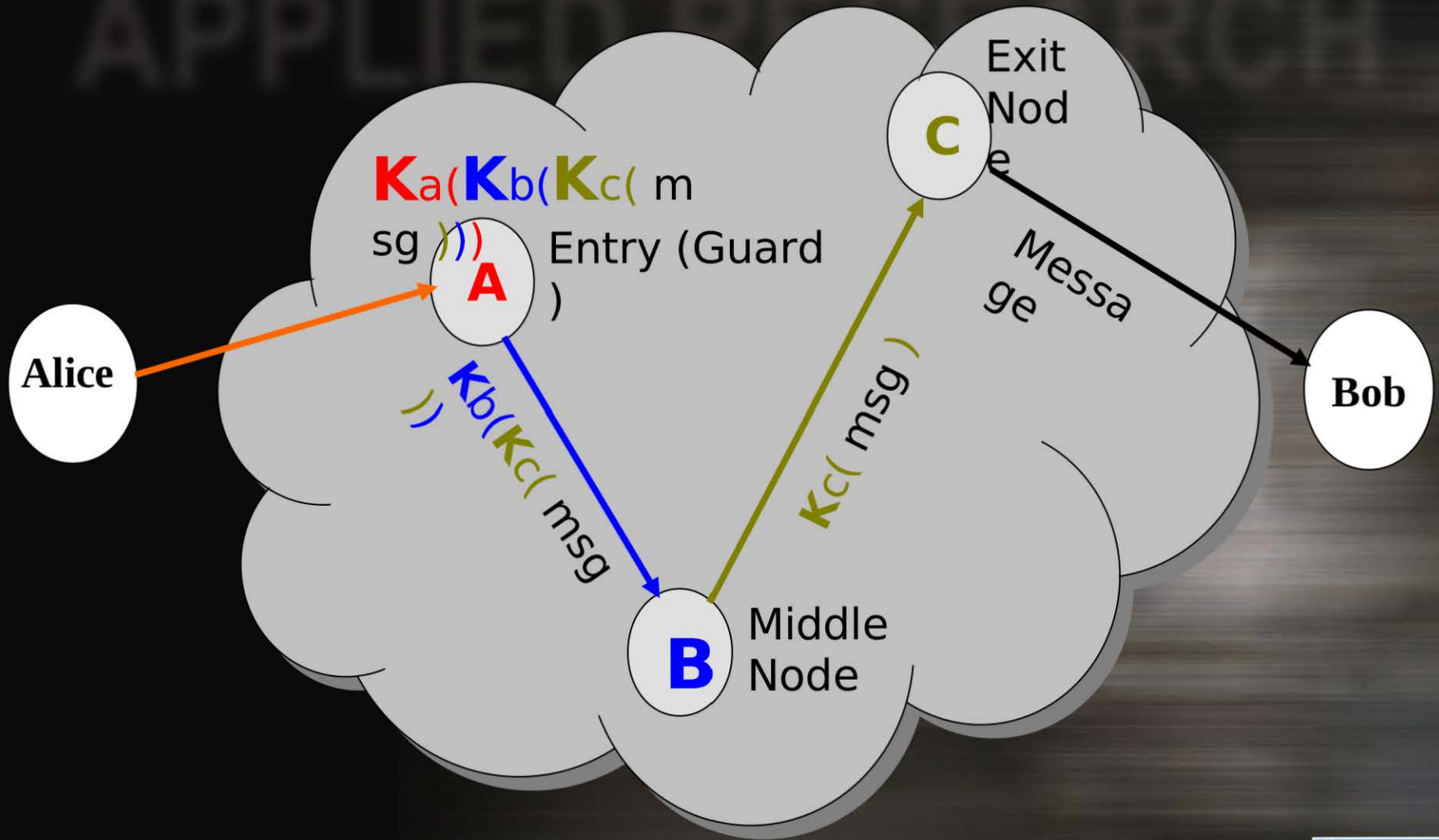
[redacted] – ICTR Network Exploitation

What is Tor?

- Tor is an implementation of 2nd generation onion routing
- Originally sponsored by the US Naval Research Laboratory
- Later became an Electronic Frontier Foundation project
- Helps to prevent network traffic analysis & surveillance
- Open network with over 2000 nodes
- Anonymity tool
- Uses multiple layers of encryption
- Multi-hop proxy

What I have done on Tor

- General Tor research
- HOMING TROLL
  - Bridge discovery capability
- Hidden Services
- Helped with a few deanonymisation techniques
- Worked with JTRIG & MCR (Maths & Crypt research)
- Provided support to OP SUPERIORITY



What is it used for?

- The Good
  - People living in oppressive countries (circumvent firewalls)
  - Access to free media instead of state propaganda
  - People can say what they want without it being linked to their public profile

- The Bad
  - Bot herders use Tor to give instructions to their bots
  - Allows paedophiles access content without linking themselves to it
  - State actors can launch attacks without being attributable
  - “Anonymous” & LULZSec

What do we see?

- Any traffic between the client & tor is heavily encrypted.
- We can only really see traffic from an exit node to a website
  - But we don't know where this traffic originated from
- Still could link up aliases though
  - 'Somebody' could still visit a dodgy forum and log in with an alias, or even send an email using a known target email address (Assuming they don't use SSL).
- Phew… at least there is some intelligence gain…. Right?

Hidden Services

- Hides the IP address of a web service
- Protects content providers by anonymously hosting content
- Publication of undesirable content
- Both client and server are anonymous to an observer and to each other

So what do we see now?

- Not much…
- All Hidden Service traffic is heavily encrypted.
- Most we can gather is that one Tor node talks to another (IP level)
- Hiding in the crowd at its best!

The dot ~~com~~ onion BOOM

- What's this .onion business?
  - TLD Tor uses to initiate a connection to a hidden service
- Example onion domain
  - 16 characters in base32 (few characters are actually missing)
  - oqznfi3tdo6nwg3f.onion
- DNS?
  - Tor uses something similar to DNS to resolve an onion domain
  - Onion domains 'resolve' to 3+ IP addresses called Introduction Points (IPT)

Pieces of the Jig-Saw

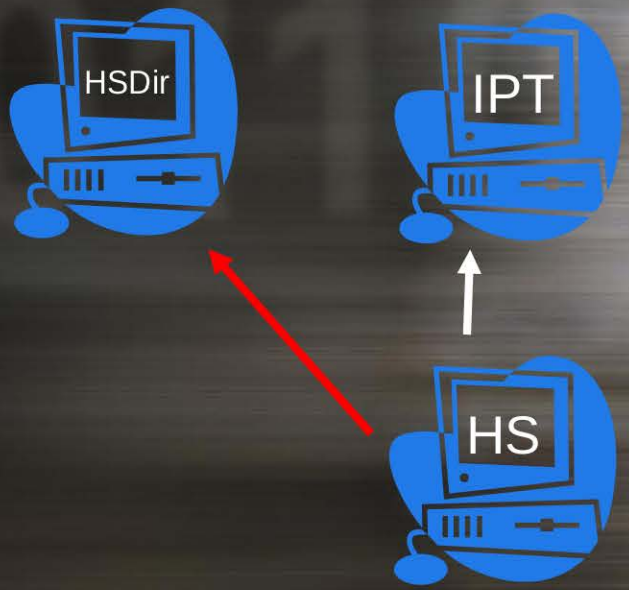
- The actual Hidden Service (HS)
  - Where the service actually originates from
- User
  - The user who wishes to access the Hidden Service
- Hidden Service Directory (HSDir)
  - A directory server that hold information on a Hidden Service
- Introduction Point (IPT)
  - Hidden Service's 'front door' / relay
- Rendezvous Point (RP)
  - Client's 'front door' / relay

Fitting it together

1. HS selects *random* IPTs
2. HS uploads descriptor to HSDir
3. Client finds out about HS
4. Client requests descriptor from HSDir
5. Client selects a random RP
6. Client contacts one IPT
7. HS replies to RP
8. RP relays between client and HS

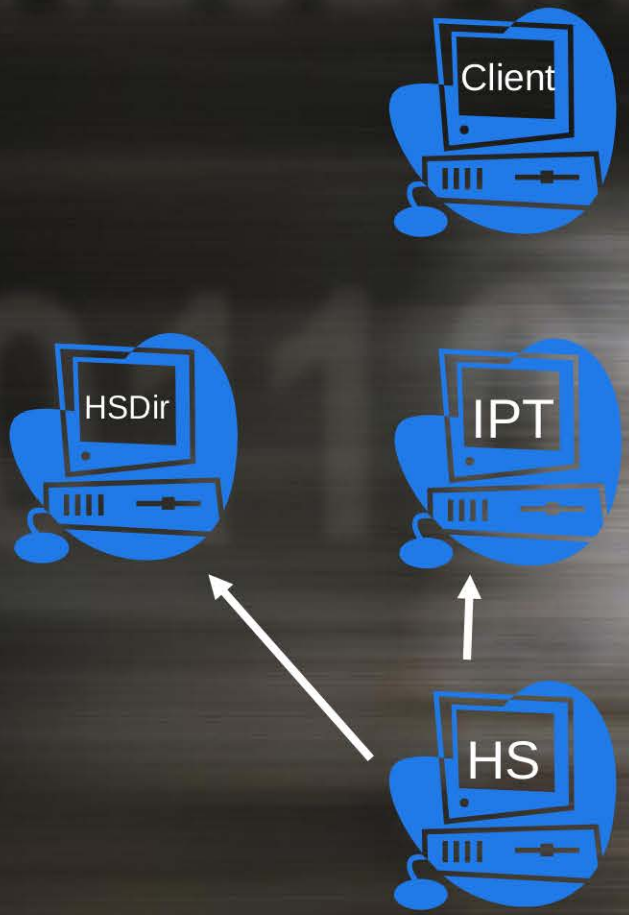
Fitting it together

1. HS selects *random* IPTs
2. HS uploads descriptor to HSDir
3. Client finds out about HS
4. Client requests descriptor from HSDir
5. Client selects a random RP
6. Client contacts one IPT
7. HS replies to RP
8. RP relays between client and HS



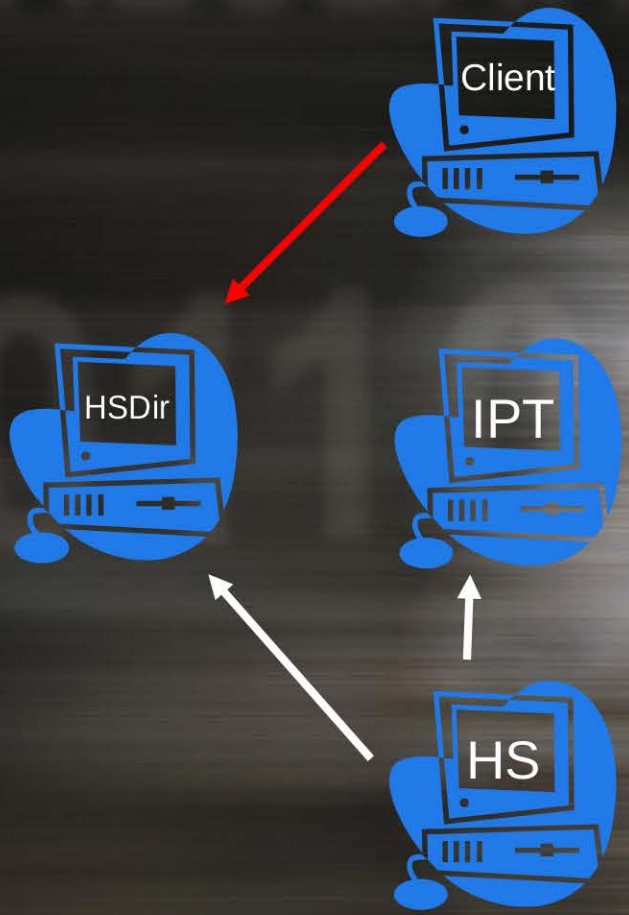
Fitting it together

1. HS selects *random* IPTs
2. HS uploads descriptor to HSDir
3. Client finds out about HS
4. Client requests descriptor from HSDir
5. Client selects a random RP
6. Client contacts one IPT
7. HS replies to RP
8. RP relays between client and HS



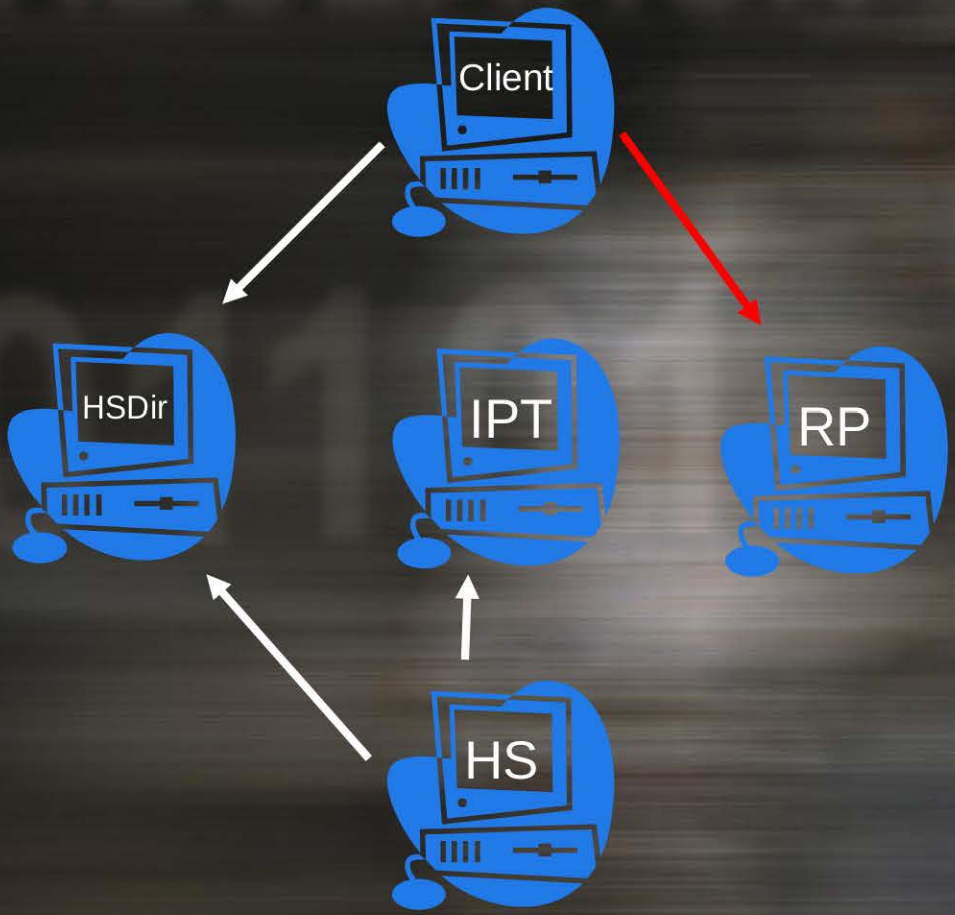
Fitting it together

1. HS selects *random* IPTs
2. HS uploads descriptor to HSDir
3. Client finds out about HS
4. Client requests descriptor from HSDir
5. Client selects a random RP
6. Client contacts one IPT
7. HS replies to RP
8. RP relays between client and HS



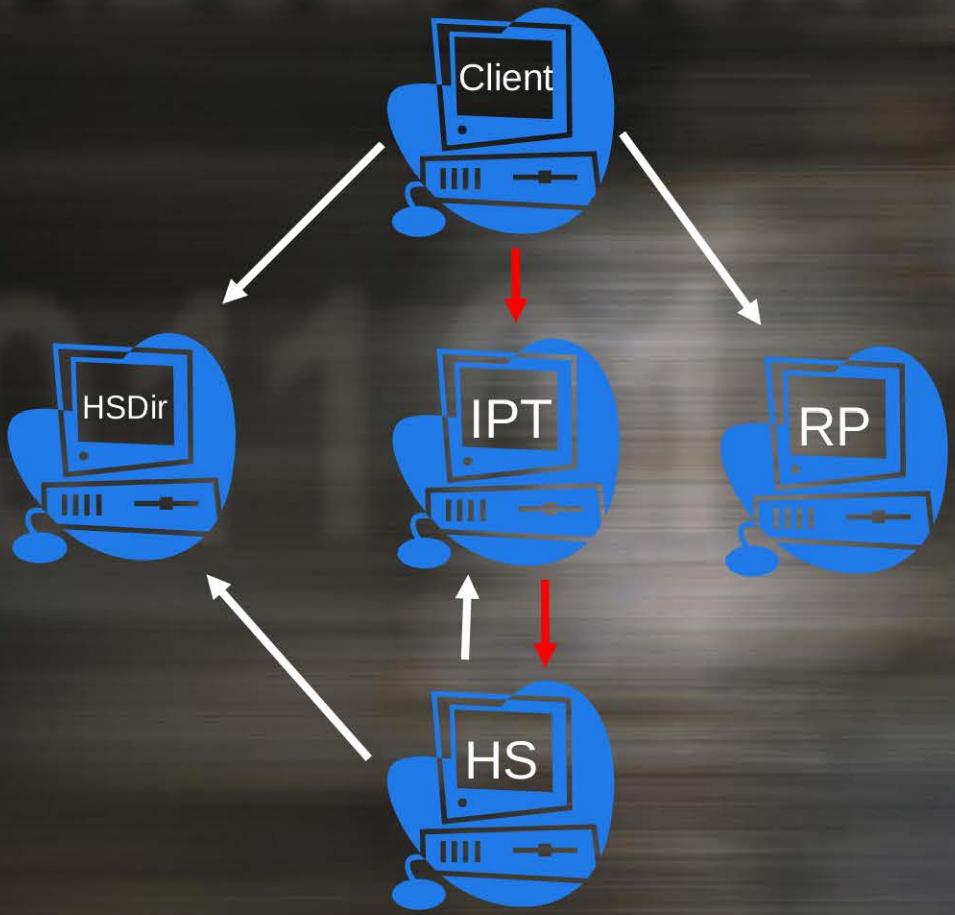
Fitting it together

1. HS selects *random* IPTs
2. HS uploads descriptor to HSDir
3. Client finds out about HS
4. Client requests descriptor from HSDir
5. Client selects a random RP
6. Client contacts one IPT
7. HS replies to RP
8. RP relays between client and HS



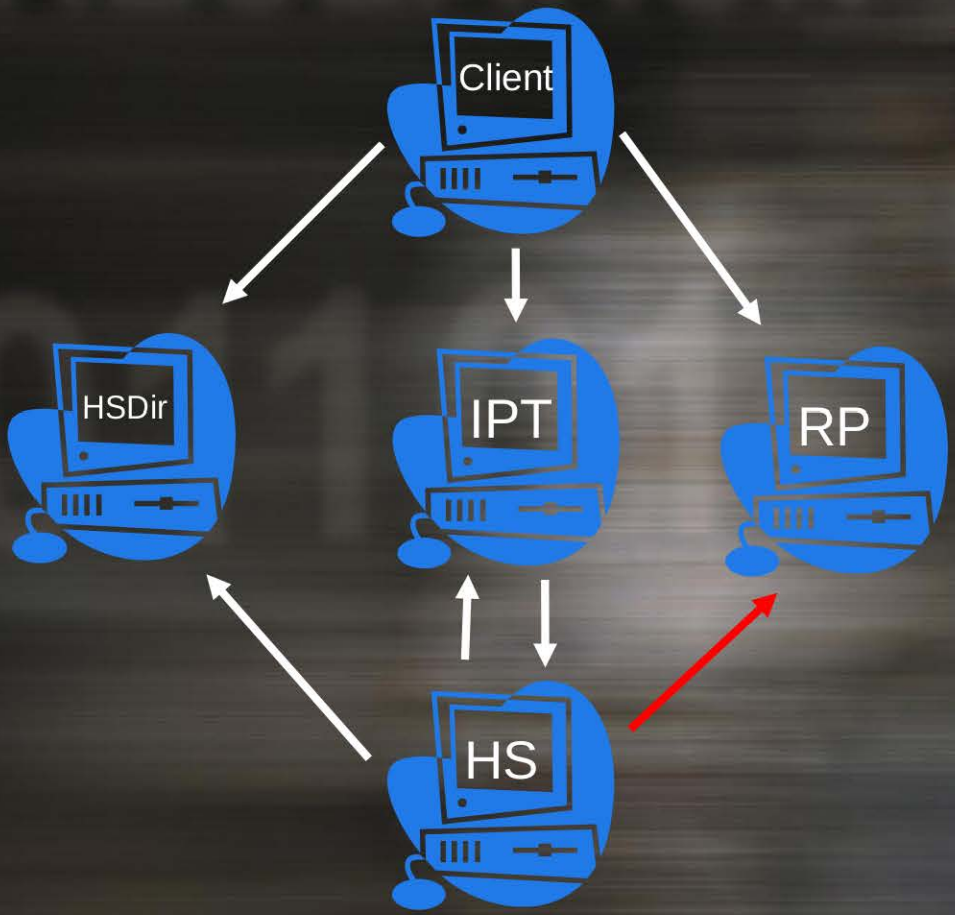
Fitting it together

1. HS selects *random* IPTs
2. HS uploads descriptor to HSDir
3. Client finds out about HS
4. Client requests descriptor from HSDir
5. Client selects a random RP
6. Client contacts one IPT
7. HS replies to RP
8. RP relays between client and HS



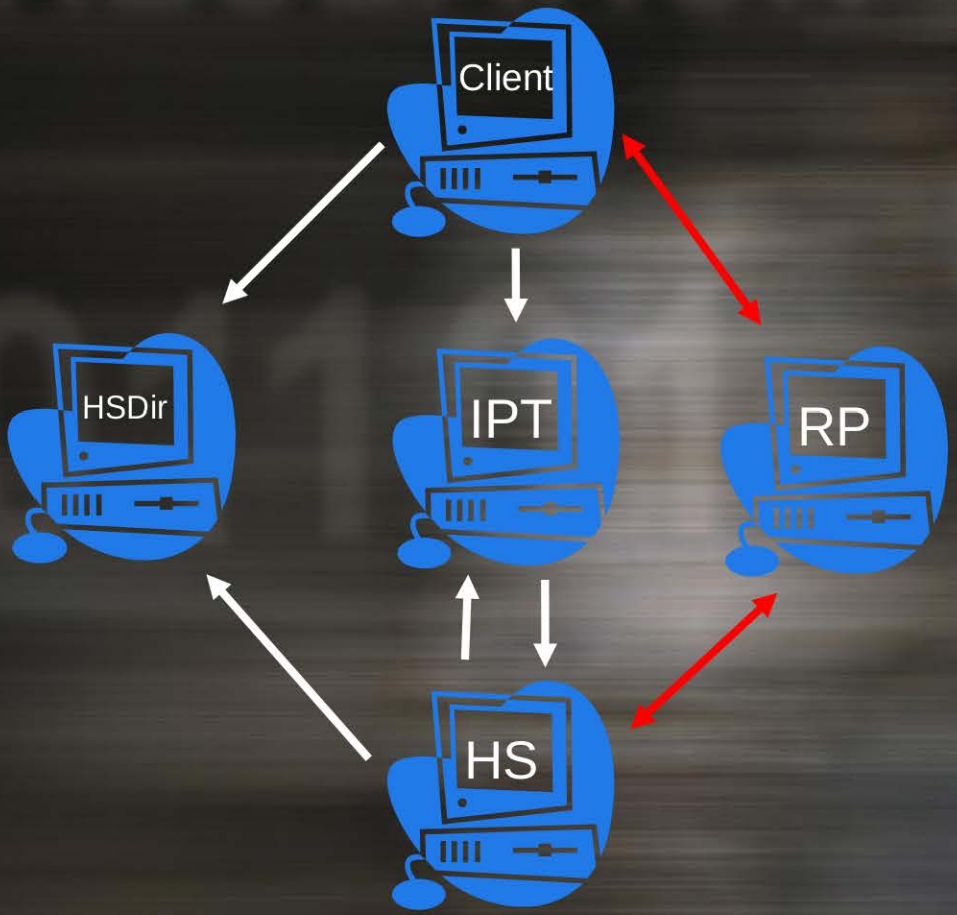
Fitting it together

1. HS selects *random* IPTs
2. HS uploads descriptor to HSDir
3. Client finds out about HS
4. Client requests descriptor from HSDir
5. Client selects a random RP
6. Client contacts one IPT
7. HS replies to RP
8. RP relays between client and HS



Fitting it together

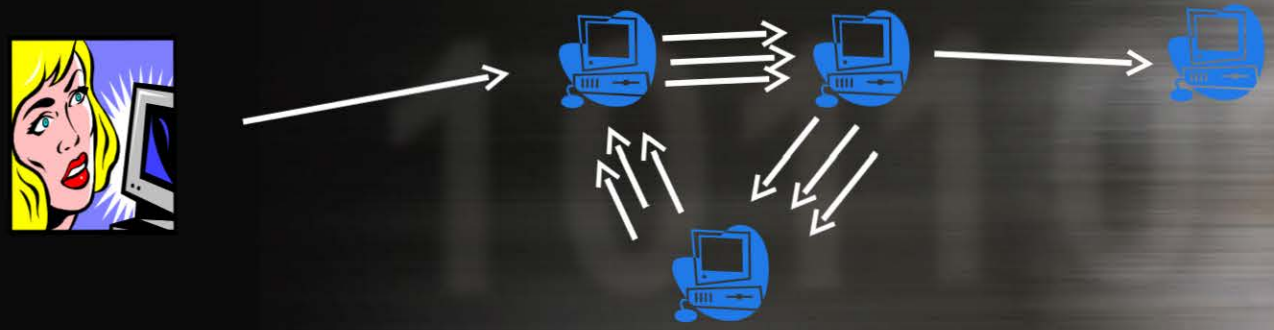
1. HS selects *random* IPTs
2. HS uploads descriptor to HSDir
3. Client finds out about HS
4. Client requests descriptor from HSDir
5. Client selects a random RP
6. Client contacts one IPT
7. HS replies to RP
8. RP relays between client and HS



Possible Exploits?

- Rendezvous Point (RP)
  - What if we owned the RP?
  - Traffic still encrypted, although only a single layer of encryption
  - Still only content, don't know who the user is or where the HS is located
  - Clients randomly select their RP so unlikely to be picked anyway
- Hidden Service Directory (HSDir)
  - If we take a HSDir down, there are still many left
  - Could potentially collect onion domains if we acted as a HSDir
- Client
  - No real way to distinguish between a Tor user accessing the web or a HS

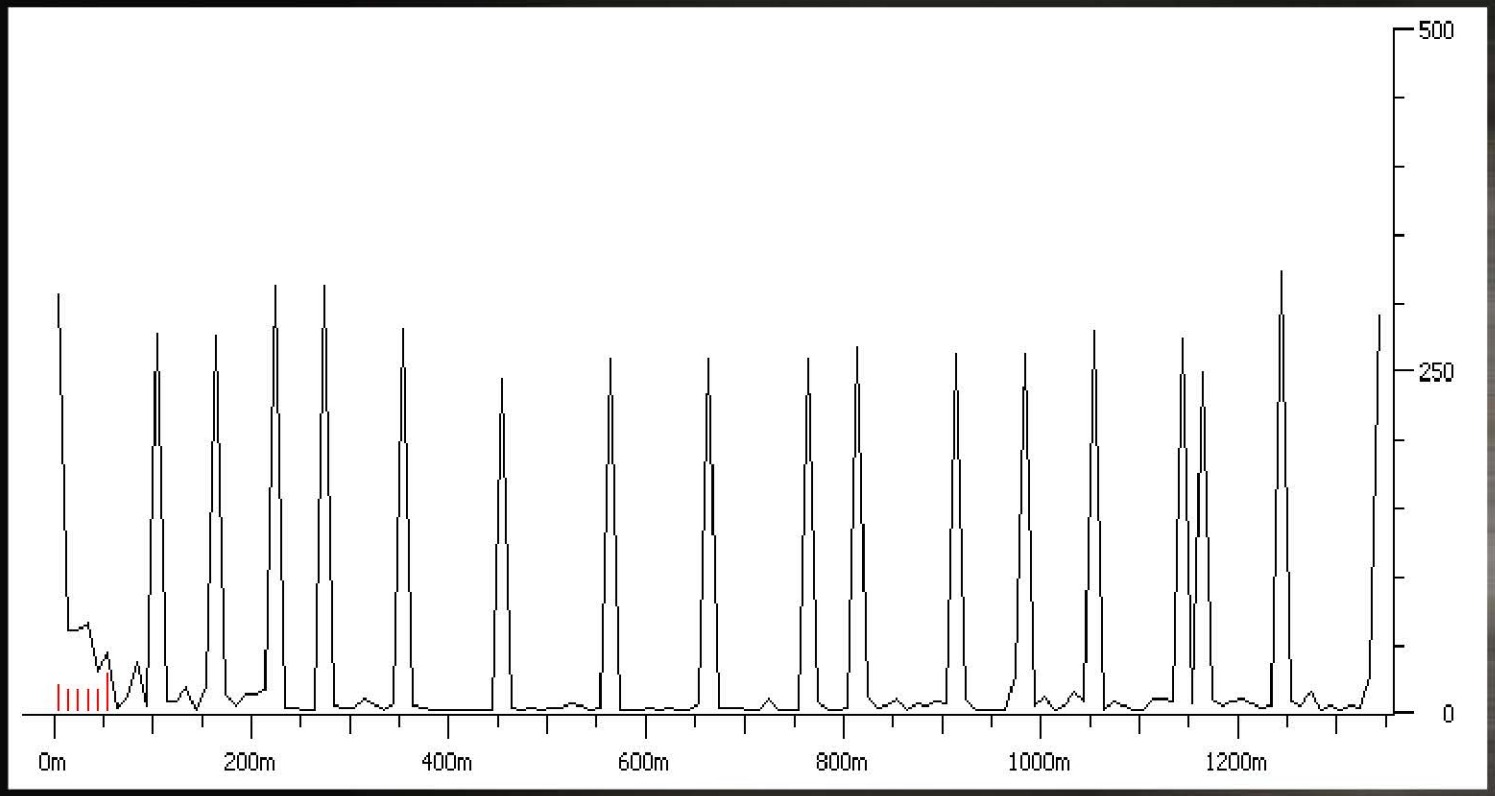
- Introduction Points (IPT)
  - All Hidden Service IPTs are listed on its descriptor (the thing that's stored on a HSDir)
  - Potential for an attack on IPTs to stop them accepting connections for the HS
  - This could be done using a 'Coil Attack'



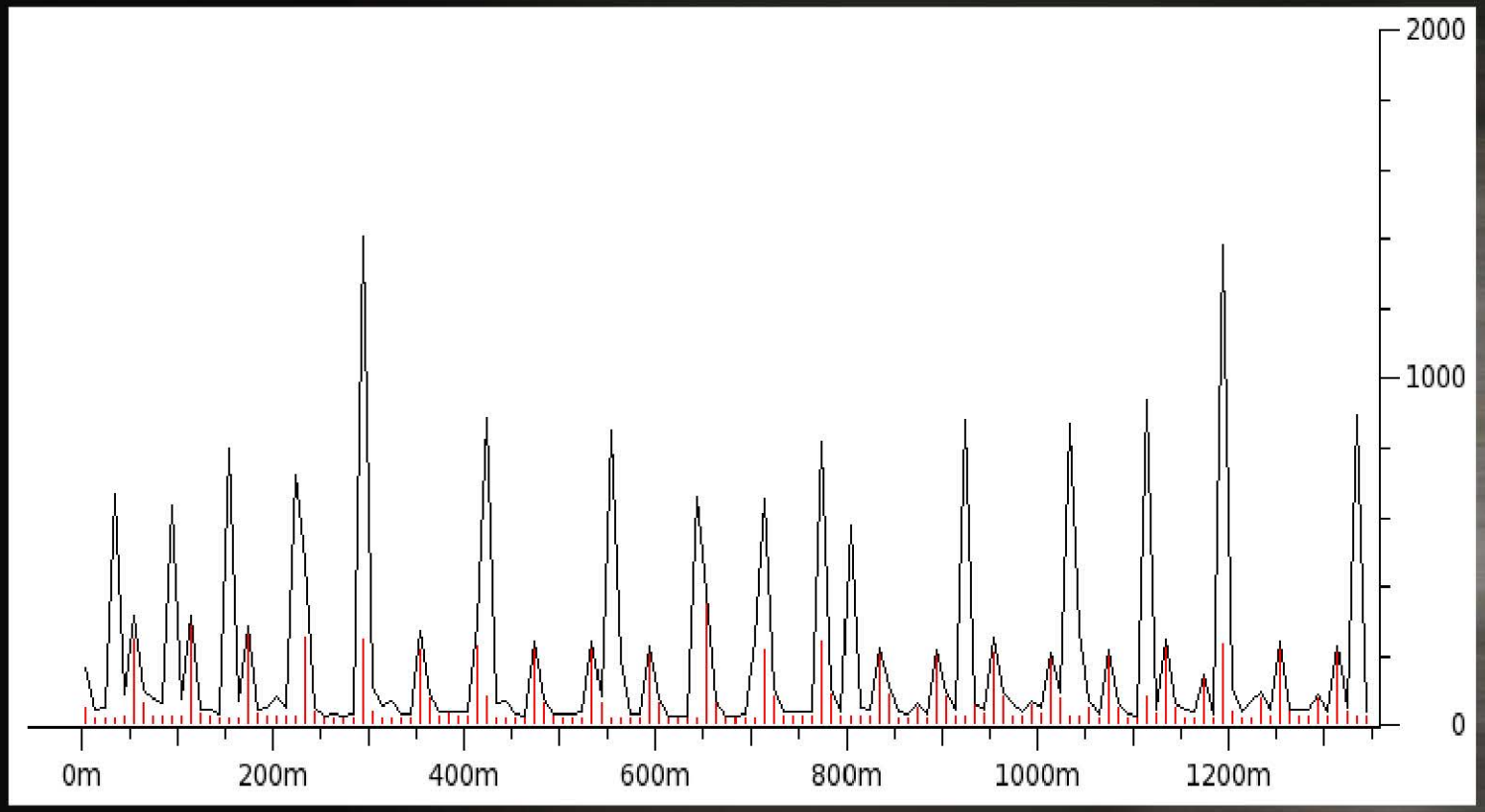
  - Doesn't stop a HS selecting another set of IPTs
  - HS can encrypt their IPTs in their descriptor (but not many do)

- Hidden Service (HS)
  - What about exploiting the HS directly?
  - Potential to identify the IP addresses hidden services
    - But cant really say which one
  - Identified a beaconing pattern from HS
  - Dependant on collection posture
  - Great for PRESTON

Idle Client Beacons



Idle HS Beacons



Summary

- Tor helps people become anonymous
- Very naughty people use Tor
- Hidden Services hide the fact web content even exits!
- Near impossible to figure out who is talking to who
- Its complicated
- Some areas for further research
- Until then… Doesn't stop us from using them

Questions?