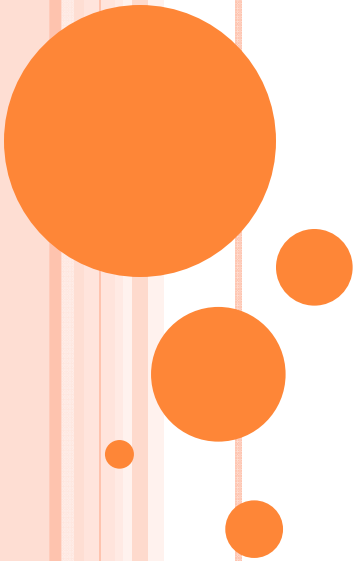


A WHITELIST APPROACH TO PROTECT SIP SERVERS FROM FLOODING ATTACKS

Eric Chen

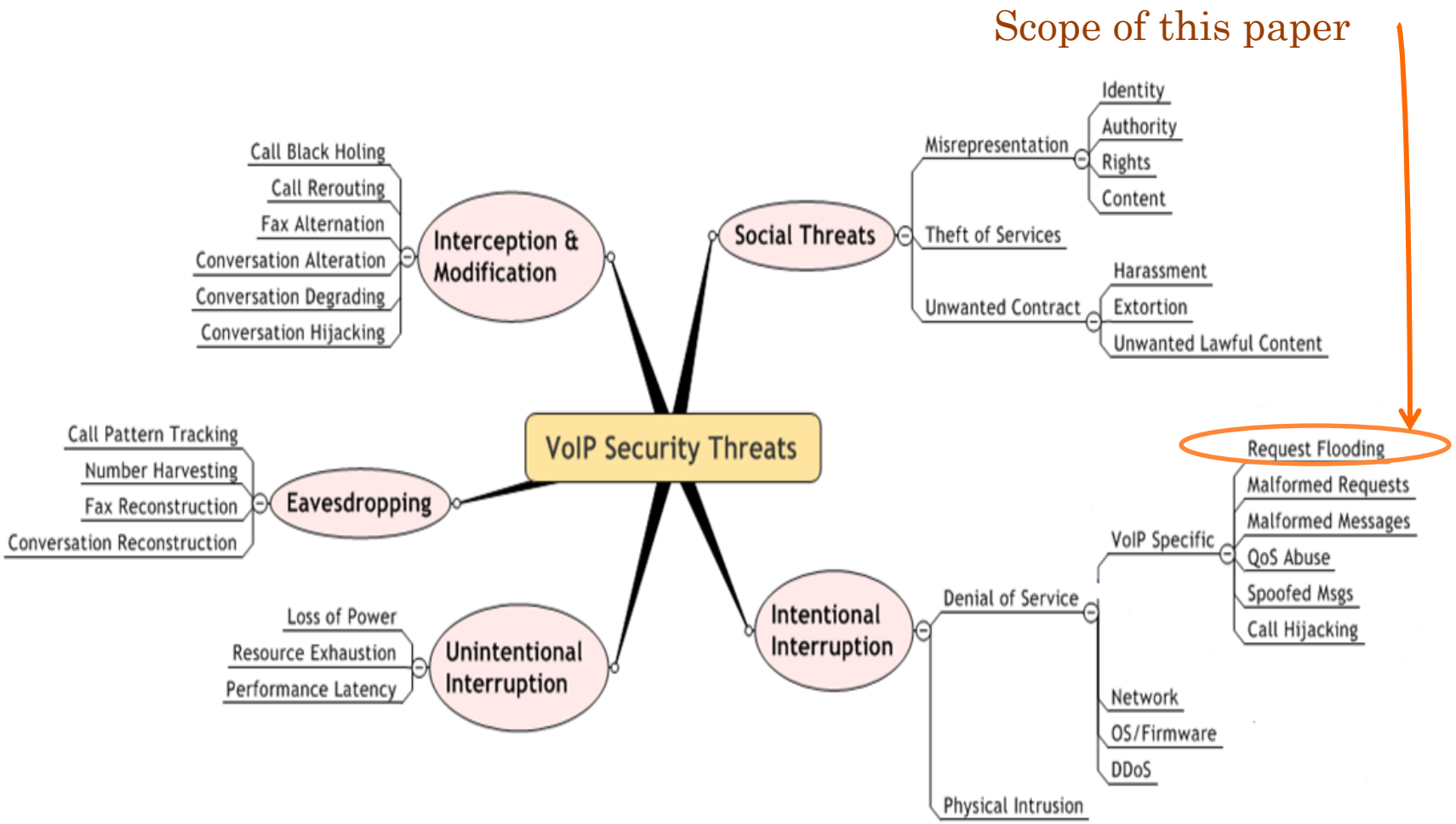
NTT Information Sharing Platform Laboratories

PRESENTATION AGENDA

1. ~~RESEARCH BACKGROUND~~
2. OUR PROPOSAL
3. EXPERIMENTAL EVALUATION

VOIP SECURITY THREATS

- VoIP services becoming potential targets of various attacks



REQUEST FLOODING

- attempts to deplete the computing resources of a SIP server by either increasing the number of SIP messages that the server has to parse or by increasing the complexity of the processing
- by overloading the computing resources, an attacker can reduce a server's capacity to handle legitimate requests

OUR RESEARCH OBJECTIVE

- Protect SIP servers from flooding attacks by maximizing their capacity to handle legitimate requests

RELATED WORK

- Luo et al. defined four types of CPU-based DoS attacks against SIP servers (basic flood, static-nonce-based flood, adaptive-nonce-based flood, adaptive-nonce-based flood with IP spoofing)
- Zhou et al. proposed history-based IP filtering to protect SIP server from these attacks by categorizing each IP address based on the number of days and the number of times it has correctly used the server

However, two questions remained to be answered:

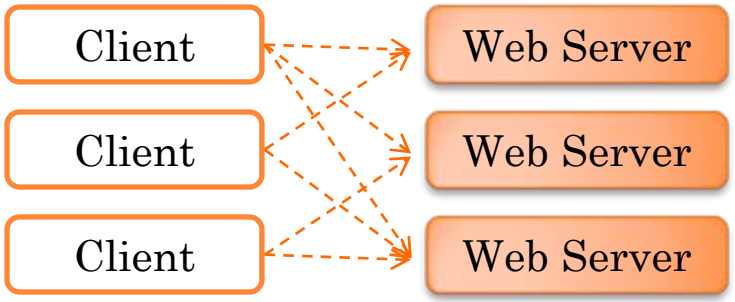
•how to determine whether a client has correctly used the server?
•how to deal with DHCP in which legitimate users may change their IP address from time to time?

PRESENTATION AGENDA

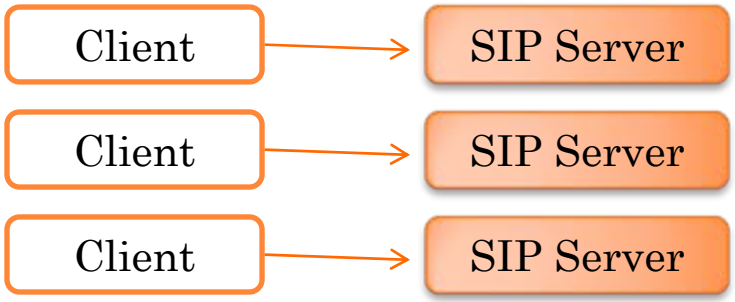
1. RESEARCH BACKGROUND
2. OUR PROPOSAL
3. EXPERIMENTAL EVALUATION

WHY IP FILTERING WITH WHITELIST IS EFFECTIVE WITH SIP?

Differences Between Web and Sip Services



- Client spontaneously visits various servers
- Connections are short-lived
- Whitelist is not effective for web services as IP addresses of regular clients are transient and unpredictable



- Client associated with a fixed SIP server
- Connections are persistent (continuous registrations)
- Whitelist can be very effective since the IP addresses of legitimate clients are predictable

SIP REGISTRATION

- SIP server accepts registration from clients and updates client information in order to perform location service for the domain it handles
- SIP clients are expected to renew their registrations frequently (usually within 1 hour) to
  - update their location information (mainly the IP address)
  - confirm that they are still actively online.

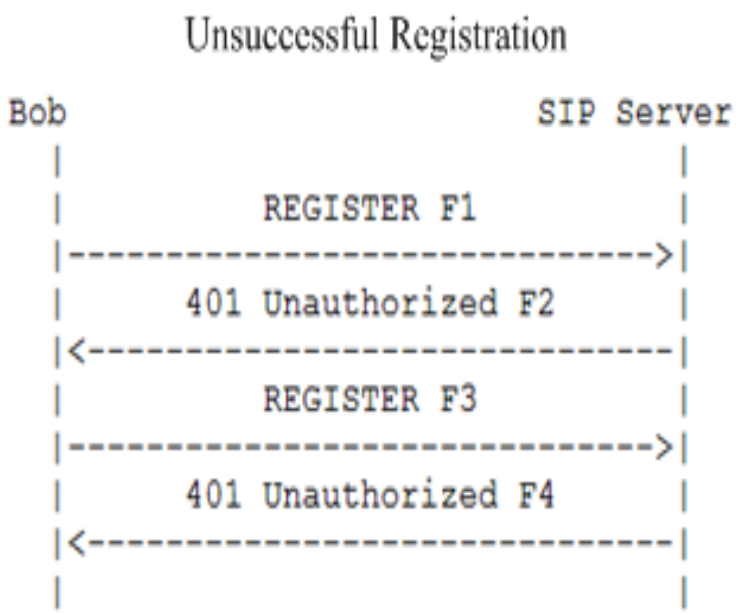
PROPOSED WHITELIST FOR SIP CLIENTS

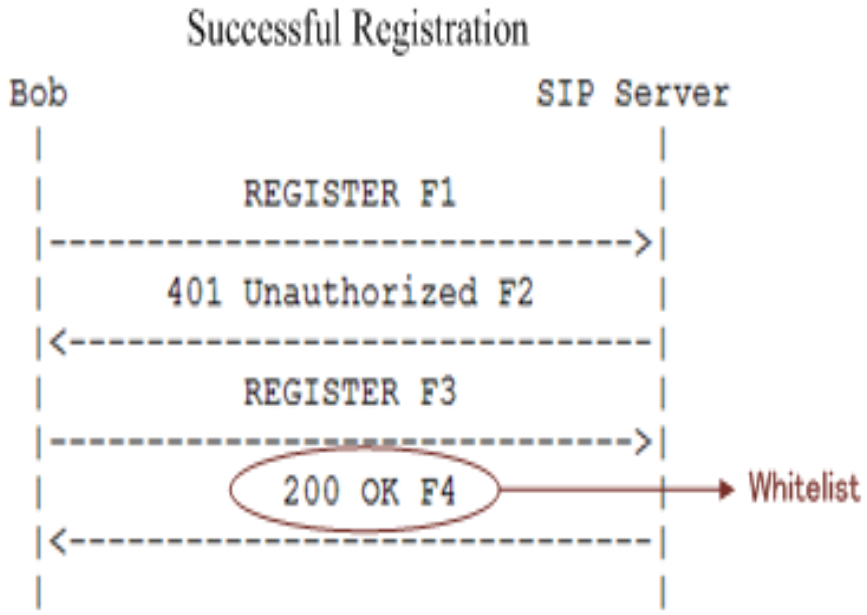
UID	*IP Address*	*Timestamp*	*Expiry (in Sec)*
012-2345-6789	1.1.1.1	2009-12-1 13:29:03	3600
012-2321-8374	2.2.2.2	2009-12-1 13:46:37	3600
012-2123-8792	3.3.3.3	2009-12-1 13:05:49	3600
....	....	....	....

- Each entry in the whitelist is added or updated using UID as the primary key every time a client performs registration.
- To keep the whitelist up-to-date, an entry is deleted if a client deregisters or fails to reregister before expiration
- SIP requests from IP addresses on the whitelist are given priority to be forwarded to the SIP server when a flooding attack occurs

HOW TO BUILD A WHITELIST?

- Directly look up the SIP database? **No**
  - Requires SIP server integration
  - Introduces extra overhead to a SIP server
- Instead, we observe traffic associated with REGISTER between a SIP server and the clients non-intrusively
- Everything we need is in a 200 OK reply to a REGISTER





PARSING A 200 OK REPLIES

- STEP 1: Scan only the first line of each message from the server. Go to Step 2 if it's a 200 OK Reply

```
SIP/2.0 200 OK
Via: SIP/2.0/TLS client.biloxi.example.com:5061;branch=z9hG4bKnashd92
 ;received=192.0.2.201
From: Bob <sips:bob@biloxi.example.com>;tag=ja743ks76zlflH
To: Bob <sips:bob@biloxi.example.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@biloxi.example.com
CSeq: 2 REGISTER
Contact: <sips:bob@client.biloxi.example.com>;expires=3600
Content-Length: 0
```

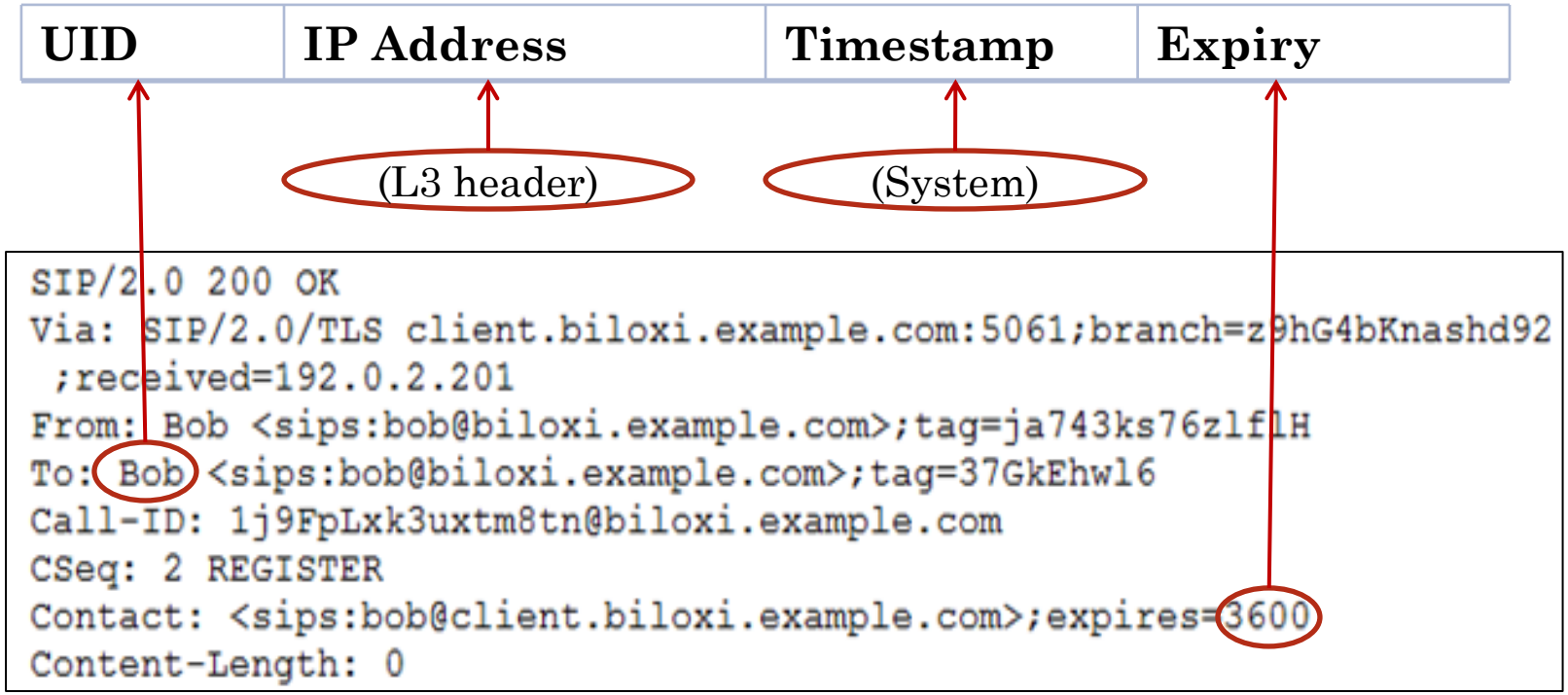
PARSING A 200 OK REPLIES

- STEP 1: Scan only the first line of each message from the server. Go to Step 2 if it's a 200 OK Reply
- STEP 2: Scan CSeq and go to Step 3 if the value is REGISTER

```
SIP/2.0 200 OK
Via: SIP/2.0/TLS client.biloxi.example.com:5061;branch=z9hG4bKnashd92
 ;received=192.0.2.201
From: Bob <sips:bob@biloxi.example.com>;tag=ja743ks76zlflH
To: Bob <sips:bob@biloxi.example.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@biloxi.example.com
CSeq: 2 REGISTER
Contact: <sips:bob@client.biloxi.example.com>;expires=3600
Content-Length: 0
```

PARSING A 200 OK REPLIES

- STEP 1: Scan only the first line of each message from the server. Go to Step 2 if it's a 200 OK Reply
- STEP 2: Scan CSeq and go to Step 3 if the value is REGISTER
- STEP 3: Update the whitelist



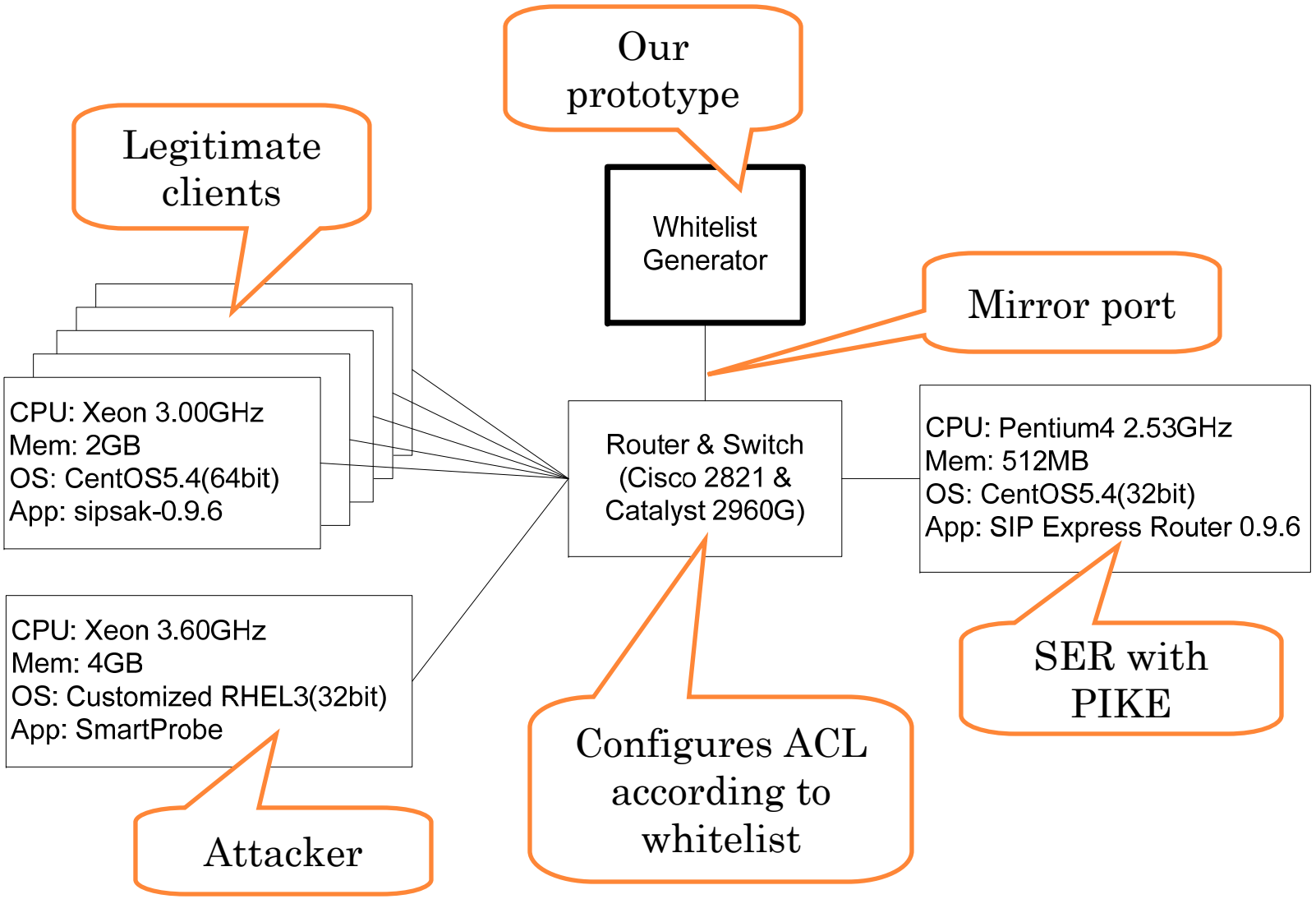
HARDENING THE DEFENSE

- Our assumption:
  - clients with valid user credentials are legitimate.
- But:
  - a botnet of compromised legitimate SIP clients with valid credentials can overcome our strategy
- Ideal:
  - combine our whitelist with a blacklist, such as keeping track of the rate of traffic coming from each IP address and blocking the packets coming from sources that exceed a predefined limit (e.g. PIKE in SER)

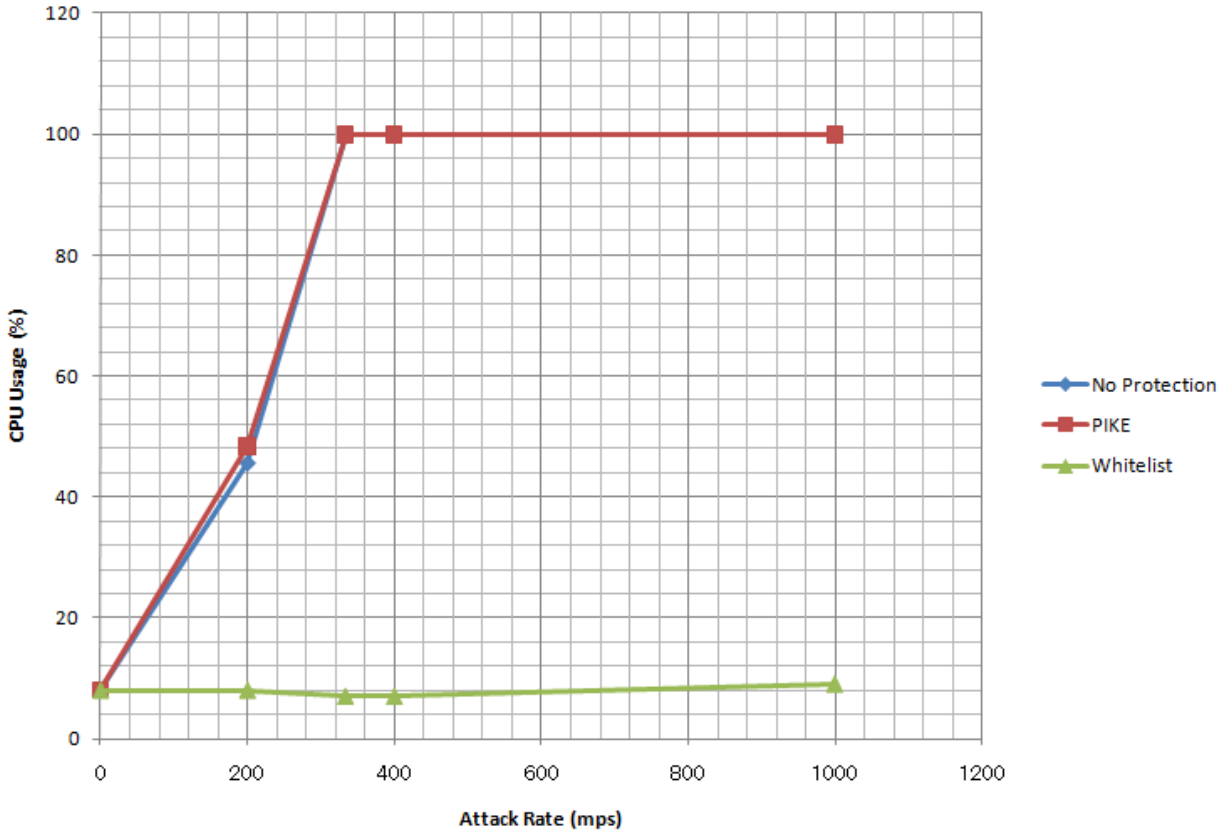
PRESENTATION AGENDA

1. RESEARCH BACKGROUND
2. OUR PROPOSAL
3. EXPERIMENTAL EVALUATION

EXPERIMENT

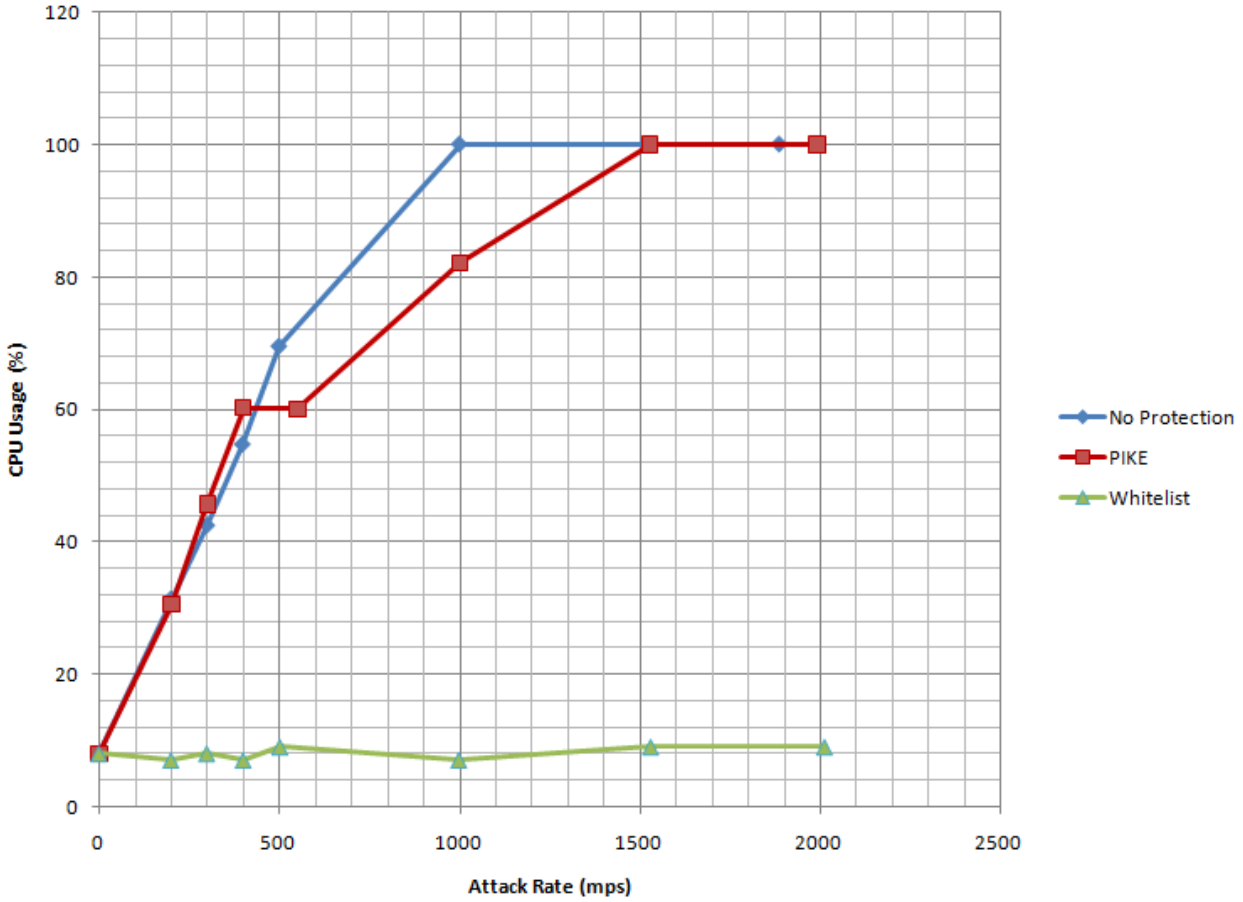

Our prototype
Legitimate clients
Whitelist Generator
Mirror port
CPU: Xeon 3.00GHz
Mem: 2GB
OS: CentOS5.4(64bit)
App: sipsak-0.9.6
Router & Switch (Cisco 2821 & Catalyst 2960G)
CPU: Pentium4 2.53GHz
Mem: 512MB
OS: CentOS5.4(32bit)
App: SIP Express Router 0.9.6
CPU: Xeon 3.60GHz
Mem: 4GB
OS: Customized RHEL3(32bit)
App: SmartProbe
SER with PIKE
Configures ACL according to whitelist
Attacker

ACK ATTACKS WITH VARYING SOURCE IP



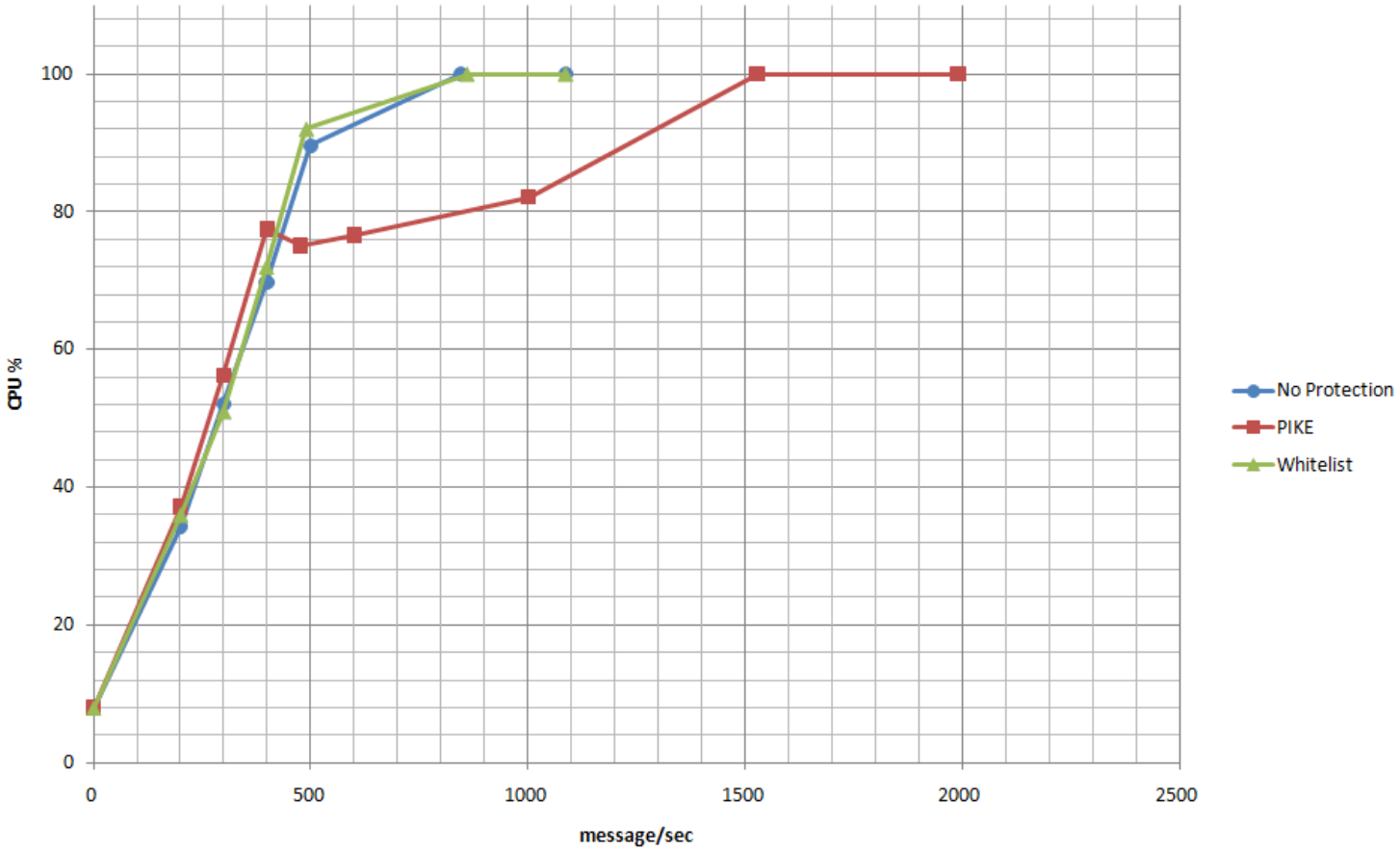
- PIKE is ineffective since attacker has no fixed addresses
- Whitelist is effective in discarding ACKs from unknown clients
- Best scenario for whitelist

REGISTER ATTACKS WITH BAD AUTH



- PIKE is somewhat effective since attackers now has fixed addresses (can further improve performance by separating PIKE from server)
- Whitelist is effective since unsuccessful registrations are ignored

REGISTER ATTACKS WITH GOOD AUTH (BOTNET)



- PIKE is somewhat effective
- Whitelist is ineffective since it includes addresses of bots
- By combining whitelist with PIKE, we can combine their strengths

CONCLUSION

- Proposed a whitelist approach to defending against flooding attacks on a SIP server.
- Relatively easy to implement as it does not require integration with a SIP server.
- Limitation in dealing with attacks from a botnet of compromised PCs with valid user credentials.
- Can overcome this limitation by combining our whitelist with a blacklist solution

Q&A