SCIDIVE: A Stateful and Cross Protocol Intrusion Detection Architecture for Voice-over-IP Environments

Yu-Sung Wu, Saurabh Bagchi
Dependable Computing Systems Lab
School of Electrical and Computer Engineering
Purdue University

http://shay.ecn.purdue.edu/~dcsl

Sachin Garg, Navjot Singh
Avaya Labs

Timothy Tsai
Sun Microsystems

Outline

- Motivation : VoIP System & Threats
- Applicability of current IDS to VoIP systems
- Design of SCIDIVE
  - Cross-protocol methodology for detection
  - Stateful methodology for detection
- Implementation
- Attack scenarios
- Future work

Motivation : Threats against VoIP System

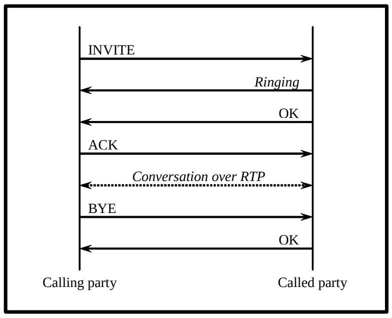
- Voice-Over-IP (VoIP) systems are gaining in popularity for carrying voice traffic over IP infrastructure
  - Economical due to convergence of data and voice
  - Useful for internal corporate communication and external inter-domain communication
- VoIP systems are vulnerable to malicious attacks
  - Traditional ones and
  - Specialized ones targeted to VoIP systems
- Protecting VoIP systems is challenging
  - Open environment
  - Employ multiple protocols
  - Systems are distributed in nature
  - Different components are under different administrative domains

Some facts about VoIP

- Voice communication between end points/terminals/clients
  - Physical VoIP phones, or
  - Software programs executing on computers
- Other entities: Gateways, Proxy servers, Redirect servers
- VoIP systems provide facilities for setting up and managing voice communication sessions
  - Protocols used are H.323 or SIP
- Media (voice data) carried using protocols such as RTP
- Health of connection monitored using RTCP or ICMP
- VoIP system is session aware (stateful)
  - A media flow comes after a successful call setup
  - The sequence number of RTP packets is monotonic
  - A hang-up event happens only if there is an existing talk session

Vulnerabilities in VoIP Systems

- Voice traffic over data network
  - Vulnerable to traditional attacks, such as DoS and authentication
  - Additionally attacks related to toll fraud, privacy, and degrading voice quality
- A major source of vulnerabilities in the signaling protocol (SIP)
  - Headers and payload sent in clear text allowing attacks such as premature call termination, redirecting calls
- Vulnerabilities in the media protocol (RTP)
  - No authentication and encryption allowing attacks such as injection of spurious packets

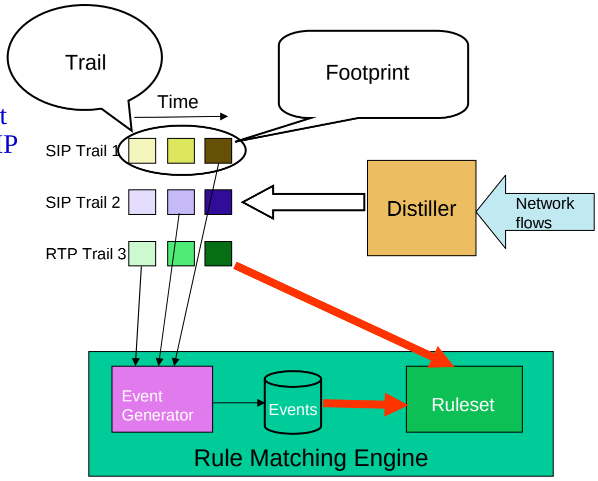


Applicability of Existing IDSs

- Current IDS’s not well suited for VoIP Intrusion Detection
- They are restricted in their ability to match patterns across multiple packets
  - *Example*: Snort’s stream4 reassembly module can only reassemble multiple TCP packets that belong to the *same* session and then apply detection rule
- They are restricted in their ability to match patterns across multiple protocols
  - Required because several attacks are based on sequences that span multiple protocols
  - WebSTAT detects attacks against web servers by correlating events from vertically layered protocols: application level (web server logs) and OS level (OS logs)
  - In VoIP systems, correlation across horizontally layered protocols is also required

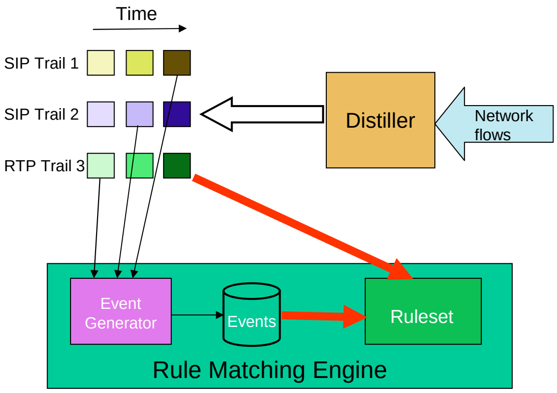
Design of SCIDIVE: Components

- Footprint
  - Protocol dependent information unit
  - For example, a Footprint can be composed of a SIP or an RTP message
- Trail
  - A set of Footprints belonging to the same session



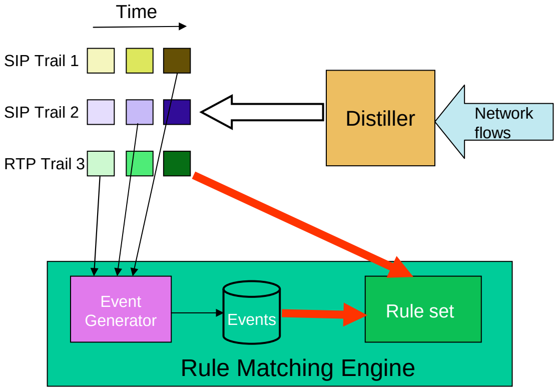
Design of SCIDIVE: Components

- Distiller
  - Translates packets into Footprints
  - Performs defragmentation, reassembly, decoding of packets
- Event generator
  - Maps footprints into events
  - Example: Map two out of order RTP Footprints into an event called ‘RtpJitter’
  - Layer of abstraction that enables efficient rule matching



Design of SCIDIVE: Components

- Rule Matching Engine
  - Triggered when events are generated
  - Works on rule set
- Rule set
  - Chiefly based on events
  - Example: Detect RTP flow (event 1) after a session is torn down (event 2)
  - Can also access information directly in trails at the cost of some efficiency
  - Example: Interested in knowing who prematurely tears down a session. Require a look at the corresponding SIP Footprint



Important Abstraction #1: *Cross Protocol Detection*

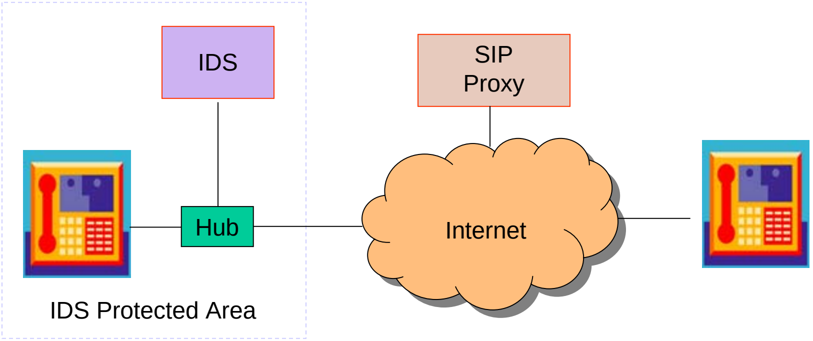
- SCIDIVE accesses packets from multiple protocols in a system to perform its detection
- Suitable for VoIP systems since it employs multiple protocols and attacks spanning multiple protocols are possible
- In SCIDIVE, cross protocol detection enabled through
  - Maintaining multiple trails for different sessions of different protocols
  - An event can be generated across multiple trails
  - A rule can be framed in terms of each of these events

Important Abstraction #2: *Stateful Detection*

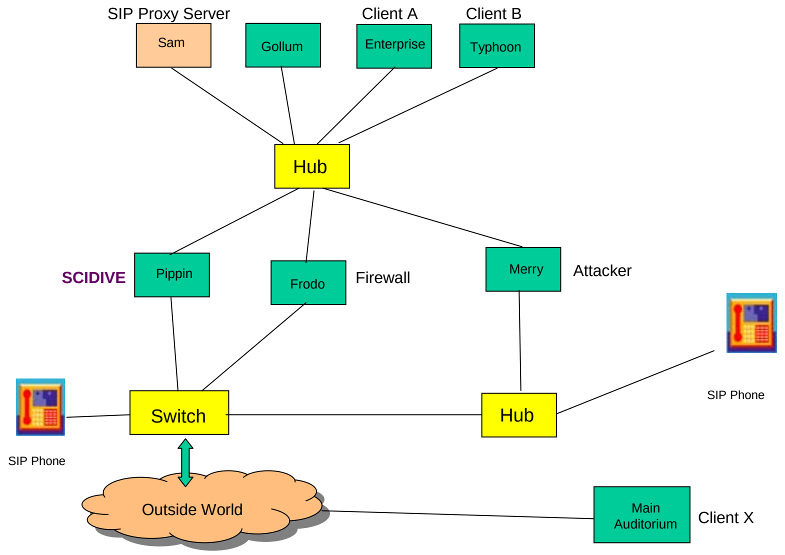
- SCIDIVE can build *relevant* state in a session and across sessions and use the state in matching for possible attacks
- Suitable for VoIP systems since components maintain considerable amount of system state
  - Client side maintains state about active connections
  - Server side maintains state relevant to billing
- In SCIDIVE stateful detection is enabled through
  - Structuring and maintaining Footprints belonging to a session in a single trail
  - Thus, state transitions of each session can be tracked

End-point based Implementation

- SCIDIVE-enabled-IDS engine sits on/close to the end-point in our implementation
- IDS engines can be deployed at multiple points – e.g., at both clients and the SIP Proxy and alert correlation done [Wu-ACSAC03]

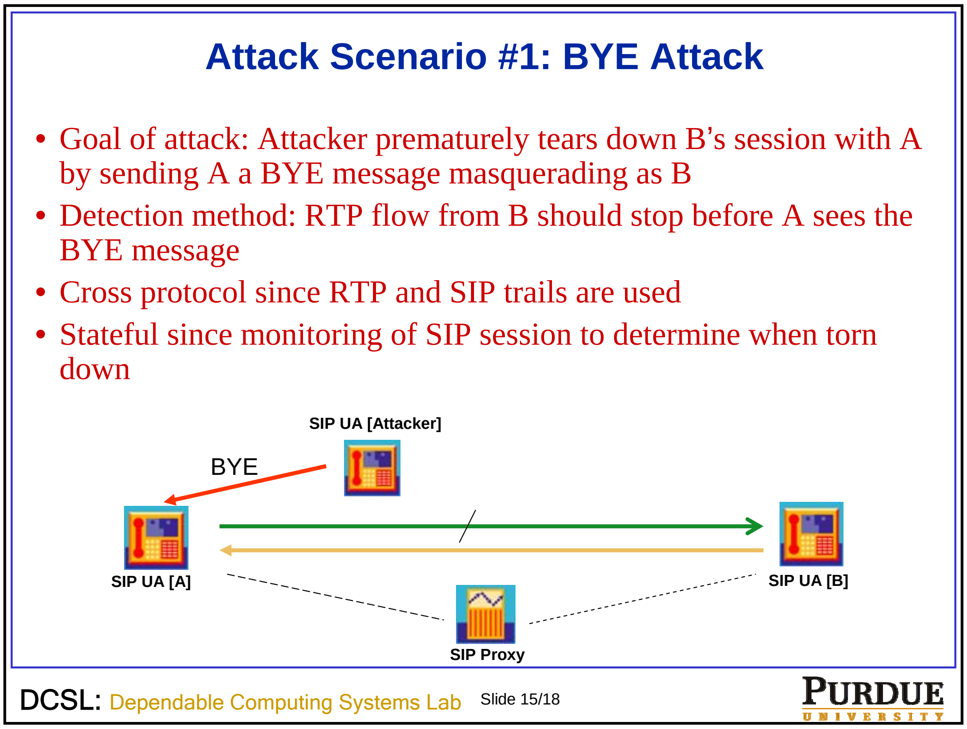


Testbed - Layout



Testbed - Details

- Protocol : Based on the trend of VoIP system development, we focus on SIP and RTP
- Proxy : Sip Express Router from www.iptel.org
- Clients :
  - Kphone from www.kde.org
  - Windows Messenger from Microsoft
  - X-Lite from www.xten.com
- Attacks created
  - BYE attack : a signaling based DoS attack
  - RE-INVITE attack : a signaling based Call Hijacking attack
  - RTP attack : a media stream based DoS attack
  - Fake Instant Messaging : a signaling based identity attack



Attack Scenario #1: BYE Attack

- Goal of attack: Attacker prematurely tears down B’s session with A by sending A a BYE message masquerading as B
- Detection method: RTP flow from B should stop before A sees the BYE message
- Cross protocol since RTP and SIP trails are used
- Stateful since monitoring of SIP session to determine when torn down

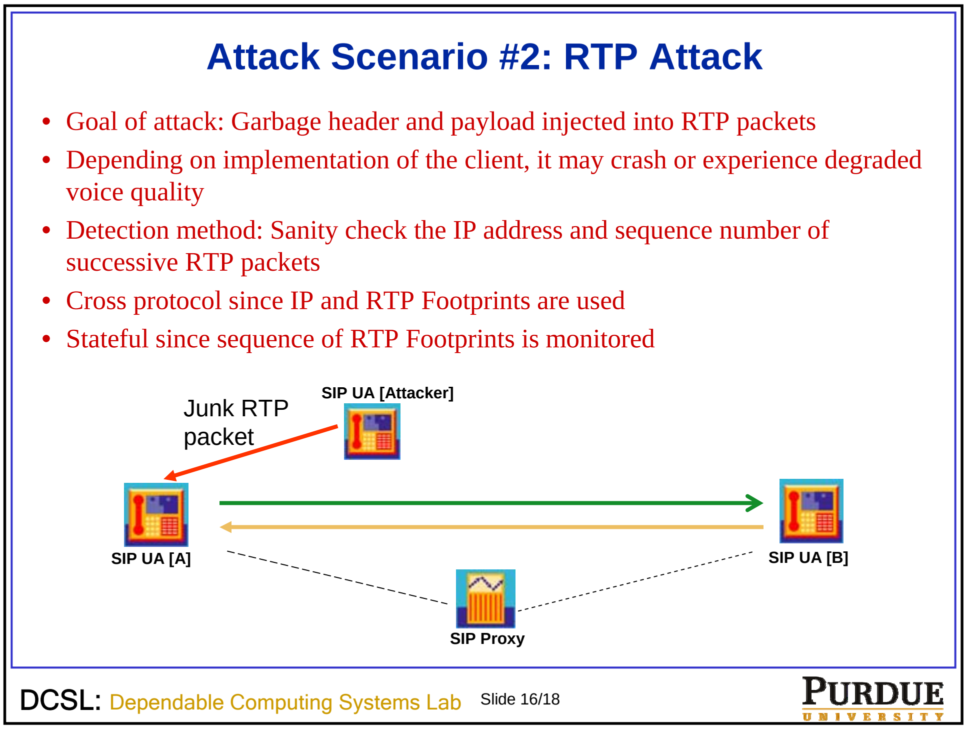
SIP UA [Attacker]

BYE

SIP UA [A]

SIP UA [B]

SIP Proxy



Attack Scenario #2: RTP Attack

- Goal of attack: Garbage header and payload injected into RTP packets
- Depending on implementation of the client, it may crash or experience degraded voice quality
- Detection method: Sanity check the IP address and sequence number of successive RTP packets
- Cross protocol since IP and RTP Footprints are used
- Stateful since sequence of RTP Footprints is monitored

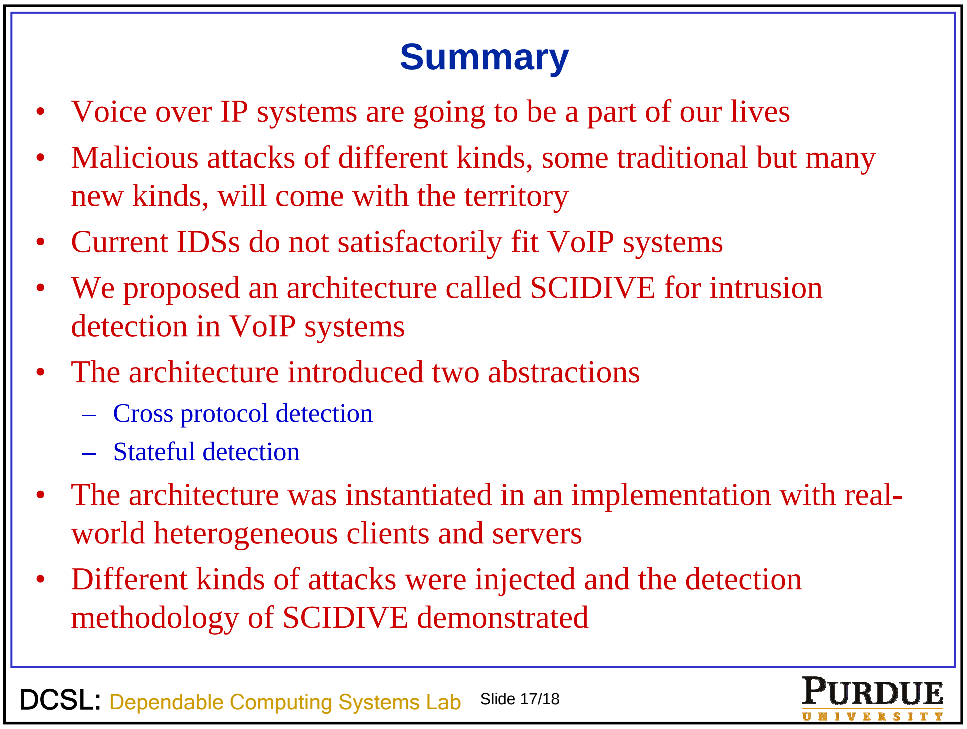
Junk RTP packet

SIP UA [Attacker]

SIP UA [A]

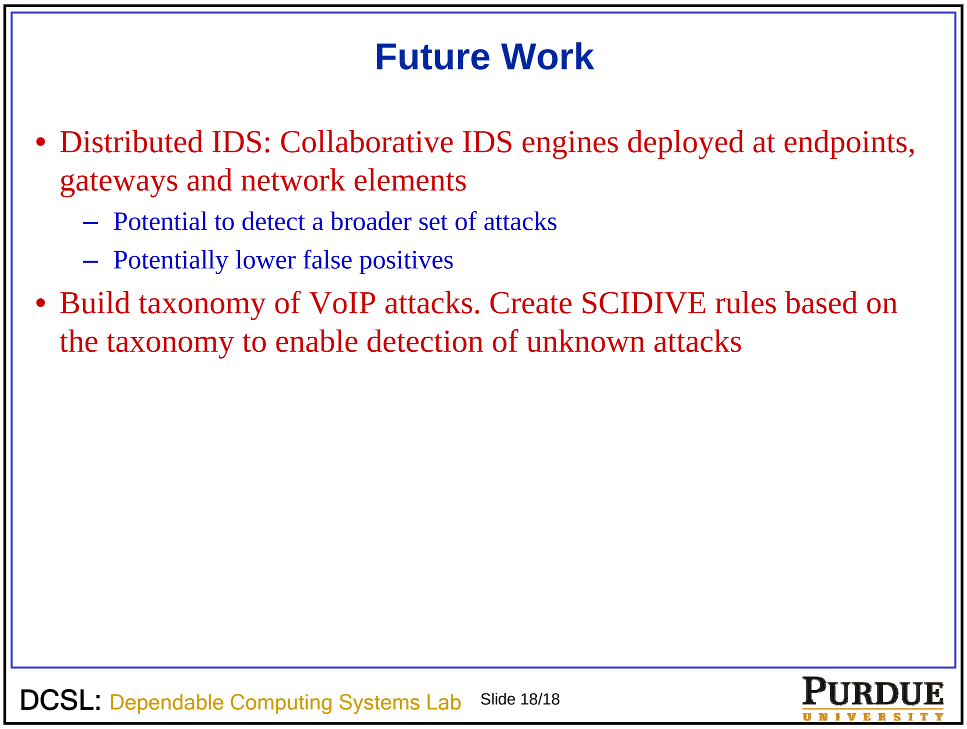
SIP UA [B]

SIP Proxy



Summary

- Voice over IP systems are going to be a part of our lives
- Malicious attacks of different kinds, some traditional but many new kinds, will come with the territory
- Current IDSs do not satisfactorily fit VoIP systems
- We proposed an architecture called SCIDIVE for intrusion detection in VoIP systems
- The architecture introduced two abstractions
  - Cross protocol detection
  - Stateful detection
- The architecture was instantiated in an implementation with real-world heterogeneous clients and servers
- Different kinds of attacks were injected and the detection methodology of SCIDIVE demonstrated



Future Work

- Distributed IDS: Collaborative IDS engines deployed at endpoints, gateways and network elements
  - Potential to detect a broader set of attacks
  - Potentially lower false positives
- Build taxonomy of VoIP attacks. Create SCIDIVE rules based on the taxonomy to enable detection of unknown attacks