Securing your SIP network with OpenSIPS
Detection, prevention and control

Vlad Paiu

OpenSIPS Project Developer

OpenSIPS Solutions

What is OpenSIPS

Open

- GPL, Open Source project
- tens of contributers, community of thousands
- used from SMB to enterprises and grade-carriers

SIP

- SIP RFC 3261 + tens of SIP extensions
- SBC, trunking, billing, ITSP, router, call center

S

- Server (registrar, proxy, LB, B2BUA, SIMPLE, NAT, apps)
- 12000 cps, 5K parallel calls, 1M subscribers
- Programmable and flexible (scripting with > 100 modules)

OpenSIPS builds and glues SIP infrastructures.

Types of Attacks

- Passive attacks
  - Make use of information from the SIP System
  - Addressed by transport encryption (signaling and media)
- Active attacks
  - Affect SIP systems operation
  - Alter system resources

Active Attacks

- **Outside attacks**
  - **Originated by non-local SIP entities**

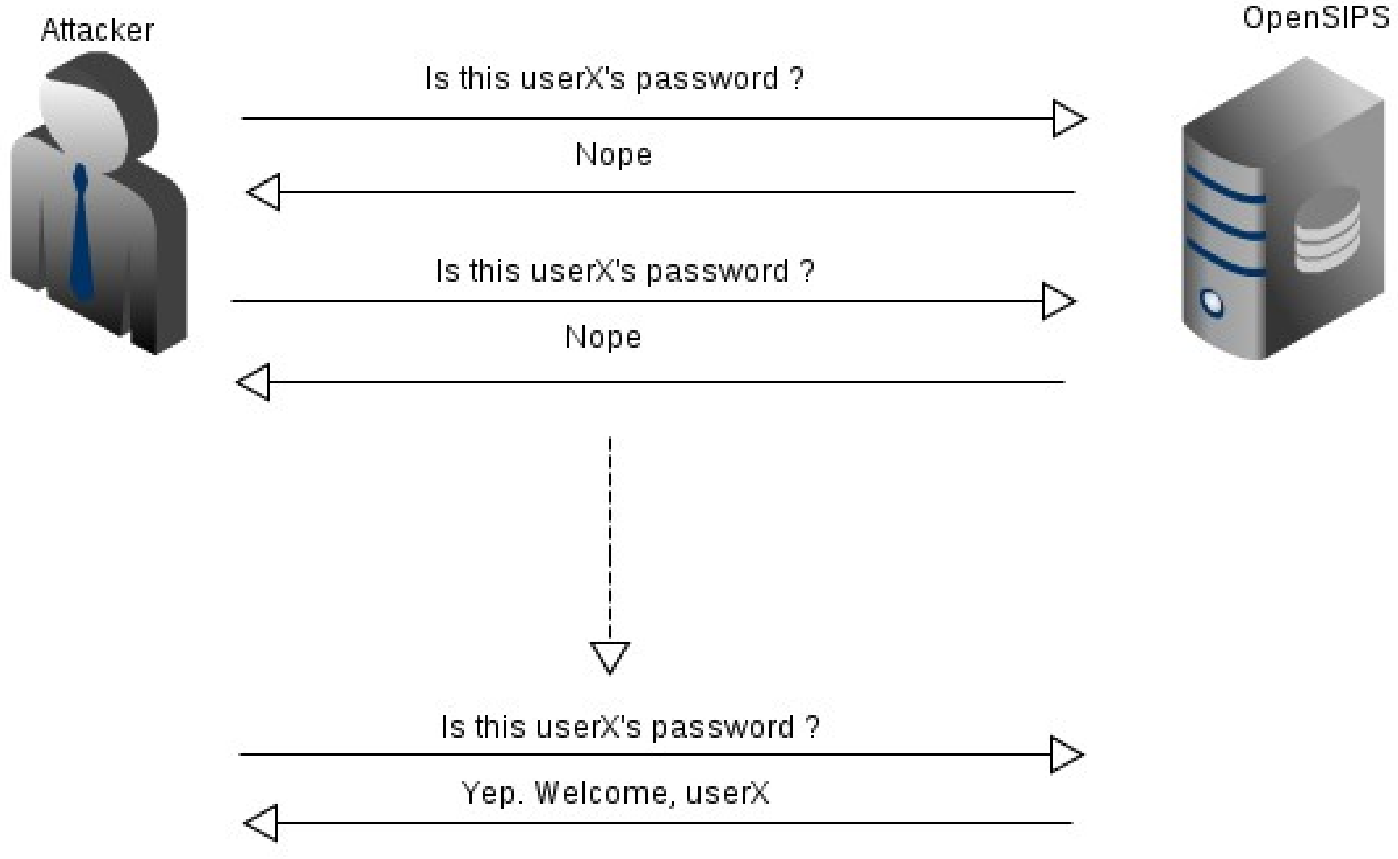
- **Inside attacks**
  - **Originated via local account – on purpose or not**
  - **Actual user or identity theft victim**

Outside Attacks

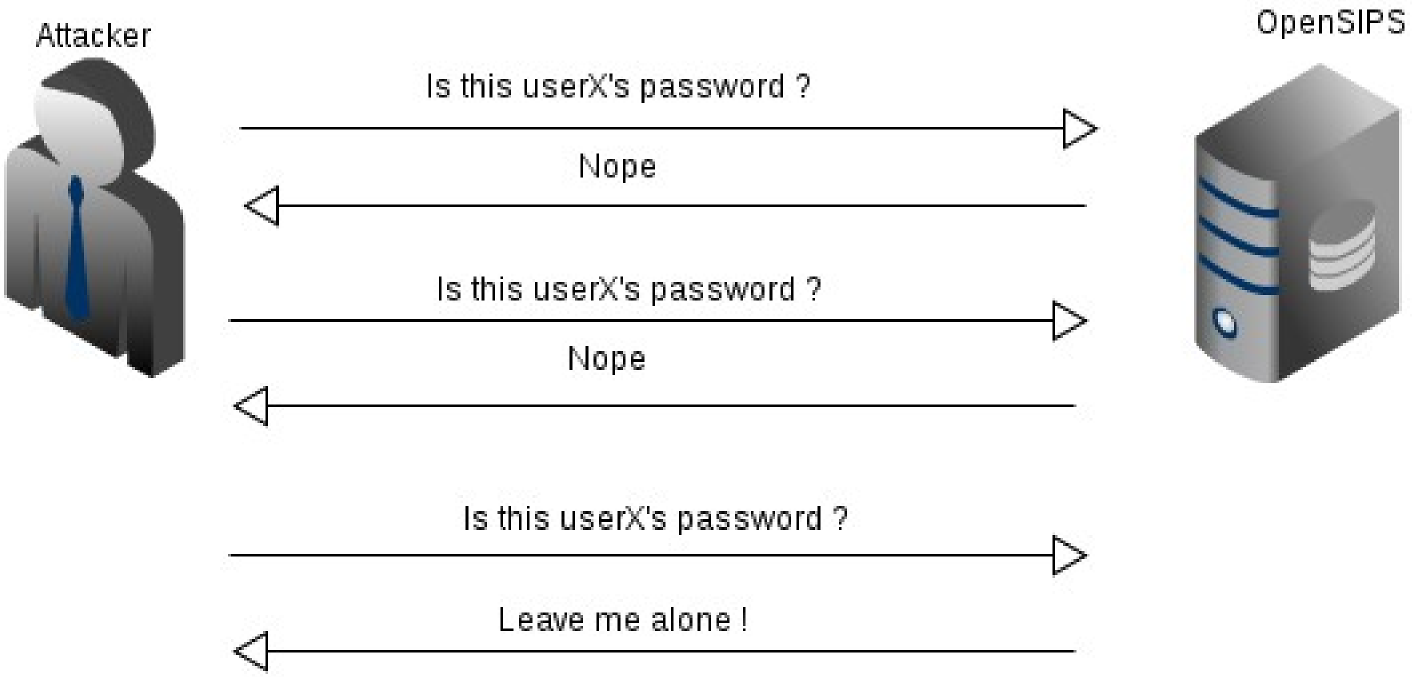
Outside attacks - Floods

- Signature Detection
  - Friendly scanner, etc
- Floods
  - Pike module
  - Check all UDP/TCP messages received
  - Event interface automagically triggered
- Rate limit
  - Ratelimit module
  - Dynamic pipes
  - Distributed

Outside Attacks - Dictionary


Attacker
OpenSIPS
Is this userX's password ?
Nope
Is this userX's password ?
Nope
Is this userX's password ?
Yep. Welcome, userX

Outside Attacks - Dictionary


Attacker
OpenSIPS
Is this userX's password ?
Nope
Is this userX's password ?
Nope
Is this userX's password ?
Leave me alone !

- **Fail2ban is a sub-optimal solution !**

Outside Attacks - Dictionary

```
www_authorize("","subscriber");
  switch ($retcode) {;
      case -3:  # stale nonce
      case -2:  # invalid passwd
      case -1:  # no such user
          xlog("Failed Auth\n");
          if ( cache_fetch("local","authF_$si",$avp(failed_no)) ) {
              if ( $(avp(failed_no){s.int}) >= 20 ) {
                  xlog("SCRIPT: SECURITY ALERT: 20 failed auth from $si\n");
                  send_reply("403","Forbidden");
                  exit;
              }
              cache_add("local","authF_$si",1,60);
          } else {
              cache_store("local","authF_$si","1",60);
          }
       default:
           xlog("Challenging\n");
           www_challenge("", "0");
           exit;
           break;
  };
```

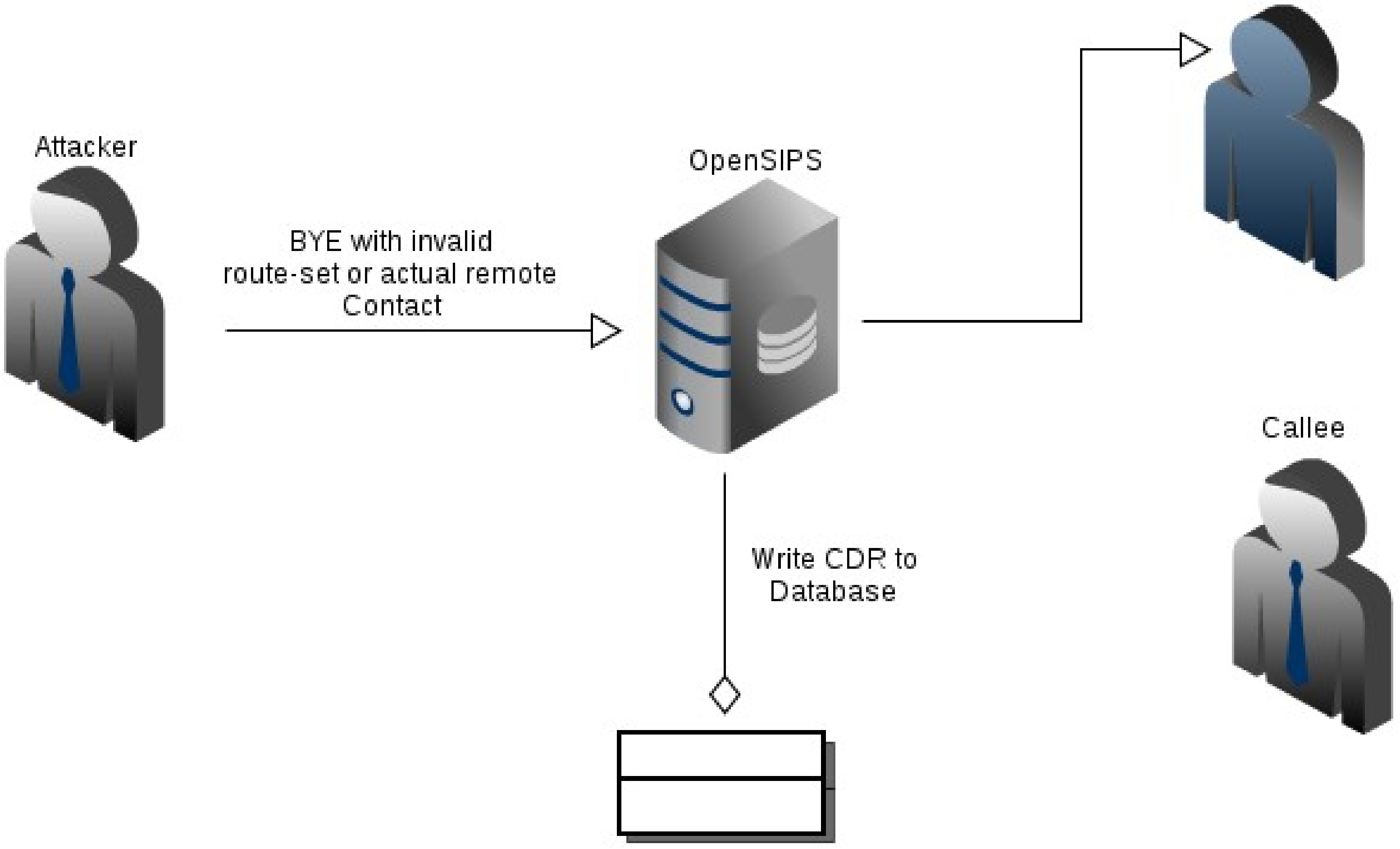
Outside Attacks - Fuzzing

- Malformed SIP packets

- sipmsg_validate() in sipmsgops module
  - Check mandatory headers are present
  - Check all header bodies
  - Check SDP body

Inside Attacks

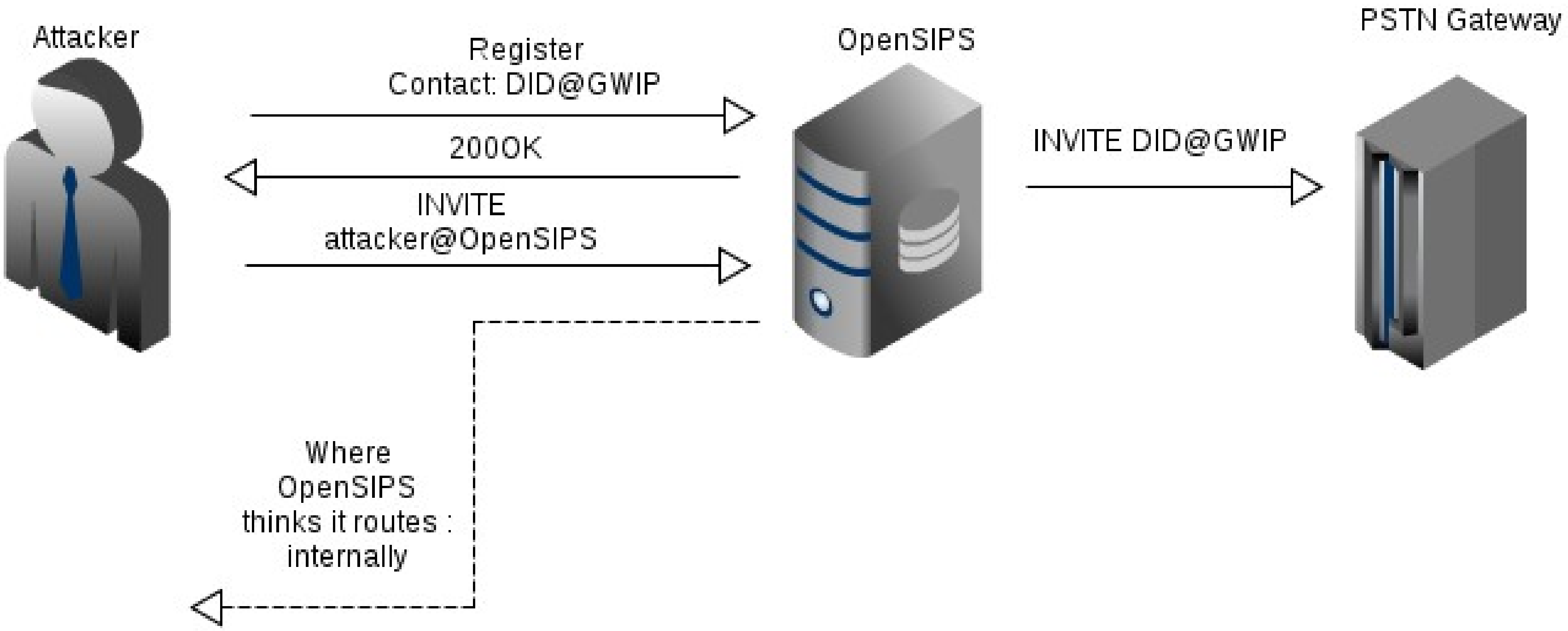
Inside attacks – SIP Injection



Inside attacks – SIP Injection

```
if (loose_route()) {
                if ($DLG_status==NULL && !match_dialog()) {
                      xlog("Unknown dialog. Might as well reject\n");
                      exit;
                }
                if (!validate_dialog()) {
                      xlog("Invalid in-dialog request\n");
                      fix_route_dialog();
                }
  }
```

Inside attacks – Register Poisoning

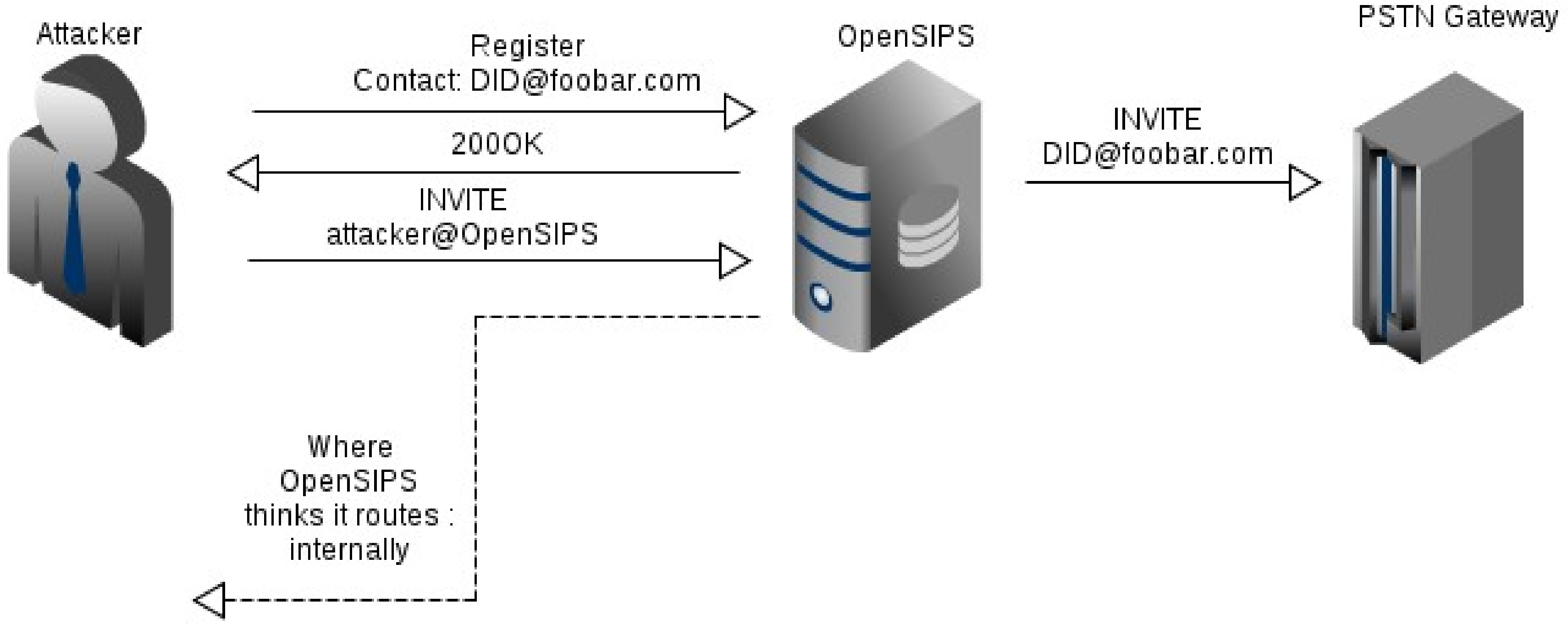


Inside attacks – Register Poisoning

```
$var(i) = 0;
while( $(ct[$var(i)])!=NULL ) {
        $var(host) = $(ct[$varv(i)]{nameaddr.uri}{uri.host});
        if ($var(host) == “GWIP” ) {
             xlog("SECURITY ALERT:  $si registering $var(host)\n");
             send_reply("476", "Contact Unacceptable );
             exit;
        }
        $var(i) = $var(i) + 1;
 }
```

Inside attacks – DNS poisoning

- **User buys foobar.com and points DNS to GWIP**



Inside attacks – DNS poisoning

```
modparam("drouting", "define_blacklist", 'gws= 0')
dst_blacklist = media:{( udp , 192.168.2.100 , 5060 , "" )
.
.
.
if (!lookup("location","m")) {
        t_reply("404", "Not Found");
        exit;
}


# make sure we do not route to gateways or media servers
use_blacklist("gws");
use_blacklist(“media”);
```

Inside attacks – Hijacked accounts

- Actual stolen accounts
  - Weak passwords
- Badly configured phones
  - Unchanged default passwords
- Exploits in the phone software
- Traffic is valid, does not look like an attack until the user starts complaining about the bill

Inside attacks – Hijacked accounts

Detect frauds as anomalies in user's dialing pattern.

- **Patterns can be :**
  - **Dynamic – Use AI algs**
    - **Learn from existing traffic**
    - **Apply learned patterns**
  - **Static - Pre-configured by the admin**
    - **When you know the traffic pattern (call-centers, etc)**

Inside attacks – Hijacked accounts

- Pattern for the volume of the calls
- Pattern for the daily schedule of the calls
- Pattern for the usual destination zones of the calls

Thank you for your attention

You can find out more at www.opensips.org

vladpaiu@opensips.org

Questions are welcome