MIC CHECK

^[1][2]$

HOMER TEAM

PROUDLY PRESENTS

IP VOICE PACKET CAPTURE FOR SECURITY AND RISK PREVENTION

INTRODUCING **HOMER 3.5**

AUTHORS:

ALEXANDR DUBOVIKOV & LORENZO MANGANI

PRESENTED AT:

KAMAILIO WORLD 2013

PRESENTATION SCHEDULE:

HOMER - Packet Capture for Security & Risk Prevention

KAMAILIOWORLD 2013

- *What's New in HOMER 3.5:*
  - SIPCAPTURE Module extension
  - HOMER API extension
  - CAPTAGENT4 & HEP3
- HOMER/Kamailio & Real-Time detections:
  - Common Security Risks with VoIP
  - Get proactive with Kamailio/HOMER
  - Examples Capture recipes
- HOMER TEAM Q&A
  - Project Roadmap & Updates
  - Out of time? Come meet us at the **sip:wise** stand

HOMER 3.5

WHAT'S CHANGED?

KAMAILIOWORLD 2013

HOMER 3.5 anticipates some of the advanced features developed for our redesigned next-generation HOMER *(Q3/4)* for our Open-Source community users, and is released today in celebration of the very 1st *Kamailio World Convention*

In this presentation we will introduce the new functionality available in **HOMER/Kamailio** and explain how to leverage the new capture logic to interact closer with your infrastructure for platform agnostic *real-time* anomaly detections.

What's new in this release?

- **Redesigned Homer API**

  All UI functions now use the new calls, designed to be 100% fruibile from external scripts

- **New webHomer functionality**

  Alarms, NRT statistics, UI Improvements (dashboard)

- **Brand new Kamailio Capture Logic**

  HOMER 3.x as our users will know, shipped out with a powerful single-function : *sip_capture();*

  HOMER 3.5+ comes with much more powerful and advanced Kamailio logic, leveraging the advanced parsing functionality and modules to perform much more than distributed capture & storing operations.

  HOMER now manages directly most part of the pre-processing conditions and database activity inside Kamailio scripting, enabling an unprecedented level of customization and feature extension, ultimately enabling the platform to perform new tasks and provide more data, some of which is perfectly suitable for real-time detections and alarming of a virtually any possible fault scenario.

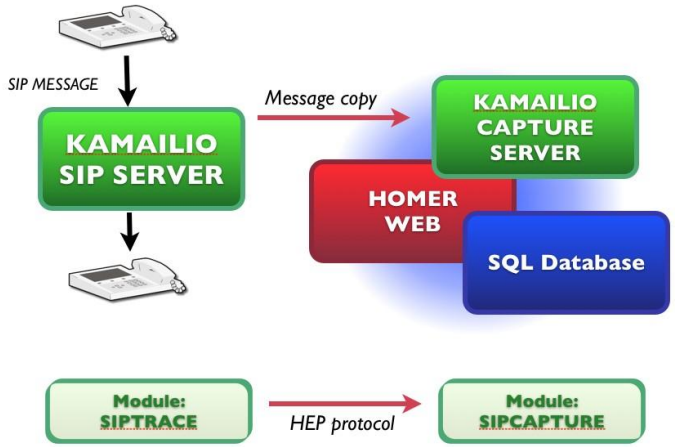
KAMAILIO 4.0

New in SIPCAPTURE module

KAMAILIOWORLD 2013

SIPCAPTURE module is a core element of the HOMER Project
Initially released in 2011, the module enables Kamailio to become a centralized capture server supporting:

- Monitoring/mirroring port
- IPIP encapsulation *(ETHHDR+IPHDR+IPHDR+UDPHDR)*
- HEP encapsulation protocol mode *(HEP v1, v2, v3)*



*What's new in **SIPCAPTURE** module:*

- **HEP version 3** implemented
- **sip_capture()** now accepts a table parameter:
  sip_capture($var(table));
- **SQL schema** fixes for PostgreSQL support
  for backcompatiblity we can use version of SQL Schema
- **X-CID** header for leg correlation is now customizable (contributor: Markus Monka)
  modparam("sipcapture", "callid_aleg_header ", "X-CID")

HOMER 3.5

HOMER'S NEW API

KAMAILIOWORLD 2013

HOMER's API has been completely redesigned to be compatible with the next-generation HOMER and now offers a clear and standardized access to all the core functionality. All of the UI Calls are now powered by the same API calls available for external scripting and polling, enabling infinite paths of interaction between HOMER and your logic/scripts

Example: Get total of OPTIONS methods in last period

REQUEST:

/api/statistic/method/total?data={"method":"OPTIONS"}

RESPONSE:

{"server":"apiserver","language":"en","status":"ok","
data":[
{"id":"11374",
"from_date":"2013-04-14 13:40:00",
"to_date":"2013-04-14 13:45:00",
"method":"OPTIONS",
"auth":"0",
"cseq":"",
"cnt":"5",
"total":"342"}
],"totalrecords":1}

PARAMETERS:

method = (can be INVITE, 200, BYE...)
cseq = can be only request
auth = 0 / 1 (0 - without auth, 1 - with)
totag = 0 / 1 (0 - without totag 1 - with)

datetime interval for stats:
from_datetime = i.e. 2012-01-01 10:00:00
to_datetime = i.e. 2012-01-01 12:00:00

HOMER 3.5

HOMER'S NEW API

KAMAILIOWORLD 2013

Example: Get ALARMS matching last period

REQUEST:

/api/alarm/data/short?data={"type":"scanner"}

RESPONSE:

{"server":"apiserver","language":"en","status":"ok","
data":[
{"id":"323",
"create_date":"2013-04-14 09:59:50",
"type":"scanner",
"total":"7195",
"source_ip":"0.0.0.0",
"status":"1",
"description":"Friendly scanner alarm!"}
],"totalrecords":1}

PARAMETERS:

type = type of alarm (defined in kamailio.cfg)
- *scanner*
- *Big messages*
- *Too many hops*
- *Loops detected*
- *Too Many 481*
- *Too Many 408*
- *Bad Requests*
- *Events Reboot*
- *Events AA*

status = 0/1 (0 - old, 1 - new)

datetime interval for stats:
from_datetime = i.e. 2012-01-01 10:00:00
to_datetime = i.e. 2012-01-01 12:00:00

CAPTAGENT 4 & HEP 3

WHAT'S NEW UNDER THE HOOD

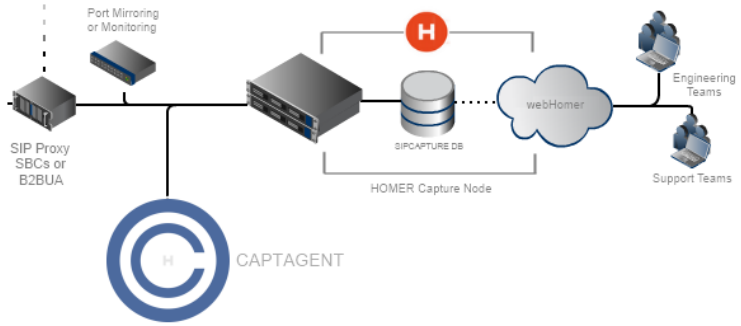
KAMAILIOWORLD 2013

HEP3 *(Homer Encapsulation Protocol)* is the glue of the HOMER Project, providing a solid and modern specification for protocol encapsulation with advanced integration and customization features suitable to support any kind of protocol; HEP3 is fully supported starting in Kamailio 4.x and is coming to other platforms soon (http://hep.sipcapture.org)

CAPTAGENT Project provides a powerful, flexible, completely modular OSS Capture-Agent framework ready for virtually *any kind of protocol* and encapsulation method. The agent can easily be extended to support new protocols and ships with full HEP3 support and universal protocol capture suitable for SIP, XMPP and many more protocols.

CAPTAGENT 4 currently supports:

- **HEP3** Implemented Features:
  - Authentication
  - Payload Compression (deflate)
  - Encryption (SSLv3/TLS)
- **UDP/TCP/TLS** Transport Supported
- **INTERNAL METHOD FILTERING** (SIP)
- **LOCAL CLI**

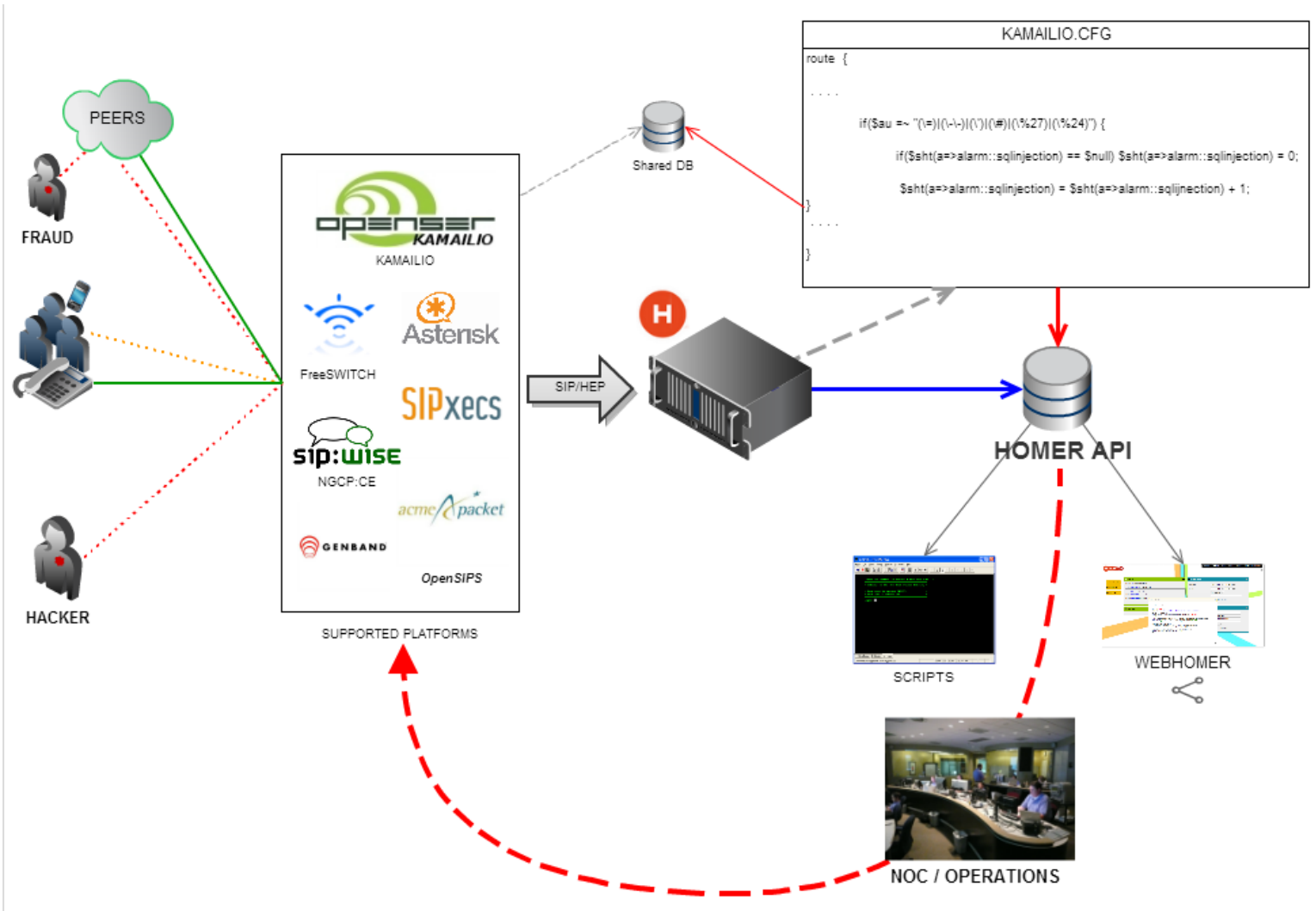


Project Homepage: **http://captagent.googlecode.com**

HOMER: Advanced

NRT Detections with Kamailio/Homer

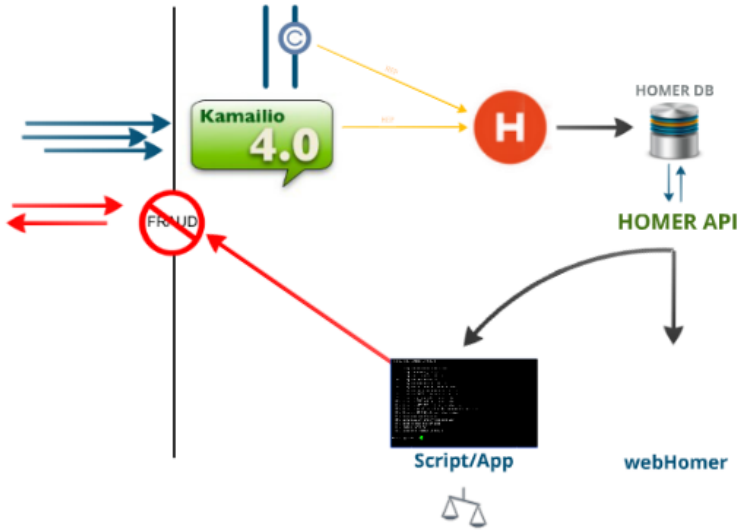
KAMAILIOWORLD 2013



HOMER: Advanced

NRT Detections with Kamailio/Homer

KAMAILIOWORLD 2013



Up Next:

- Some common Attacks
- Example Detections Scripting

HOMER 3.5 and **Kamailio 4** together allow platform agnostic real-time detections of suspect activity - such information can be a precious resource to use for risk prevention and self-defense.

Detections can focus on any desired event, covering attacks, probing/scanning, as well as fraud.

For the purpose of this presentation, we will look at some example security scenarios common to all and propose some approaches to detect and react to attacks using packet capture and **HOMER 3.5**

NOTE: This is NOT meant to be a study on VoIP attacks - just examples used for detections in Homer/Kamailio

A BRIEF OVERVIEW

of some common attacks in VoIP

DDOS ATTACKS:

- FLOOD (many requests i.e. INVITE, REGISTER, OPTIONS, NOTIFY..., SPIT (Spam over Internet Telephony)
- DNS (Via, Contact, Ruri have fake DNS hosts)
- AMPLIFICATION (forking loops)

APPLICATION LAYER ATTACKS:

- SIP SQL INJECTIONS
- PARSER ATTACKS - Malformed packet attacks
- IP Spoofing with RECORD Route, Via.
- Remote manipulation to SIP phone through Events and special headers

HIJACKING:

- SESSION TEARDOWN (SIP CALLs Termination with a “BYE” message)
- REDIRECTION CALL HIJACKING (302, REFER)
- PASSWORD HIJACKING (brute force)

SCAMMING/FRAUD:

- CALL BACK FRAUD - Call-ID Spoofing i.e (premium numbers)
- RESOURCE ENUMERATION / MAPPING

RE-NOTE: This is NOT meant to be a study on VoIP attacks - just examples used for detections in Homer/Kamailio

CAPTURE COOKBOOK

DDOS ATTACKS:FLOODING

KAMAILIOWORLD 2013

FLOODING EXAMPLE

TYPE: MANY REQUESTS (INVITE, SPIT)

HOW: counting on received methods, check for known scanners tools like sipvicious *(sometimes its as simple as checking user-agent)*

```
route {
        if($ua =~ "(friendly-scanner|sipvicious)") {
                if($sht(a=>alarm::ua::scanner) == $null) $sht(a=>alarm::ua::scanner) = 0;
                 $sht(a=>alarm::ua::scanner) = $sht(a=>alarm::ua::scanner) + 1;
         }
        ....
         if (is_method("INVITE")) {

                        if (has_totag()) {
                              if($sht(a=>method::reinvite) == $null) $sht(a=>method::reinvite) = 0;
                              $sht(a=>method::reinvite) = $sht(a=>method::reinvite) + 1;
                        }
                        else {
                              if($sht(a=>method::invite) == $null) $sht(a=>method::invite) = 0;
                              $sht(a=>method::invite) = $sht(a=>method::invite) + 1;
                              if($adu != $null) {
                                    if($sht(a=>method::invite::auth) == $null) $sht(a=>method::invite::auth) = 0;
                                    $sht(a=>method::invite::auth) = $sht(a=>method::invite::auth) + 1;
                              }
                        }
         }
        .....
}
```

capture cookbook

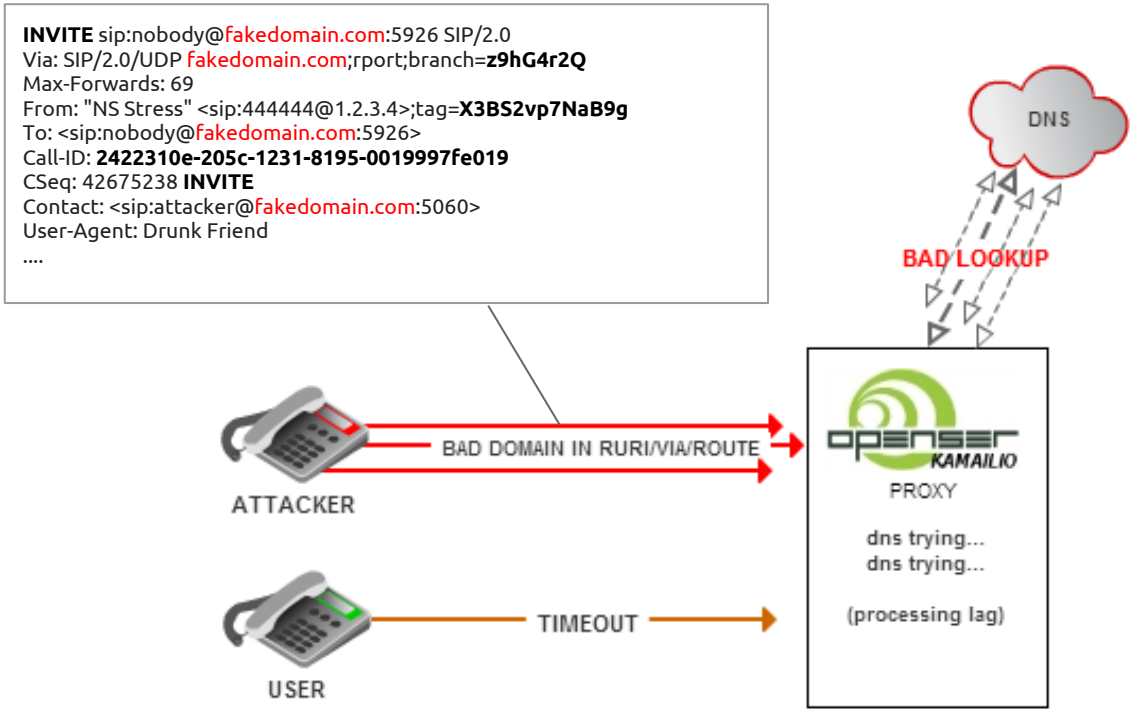
DDOS ATTACKS:

KAMAILIOWORLD 2013

DNS ATTACK EXAMPLE

TYPE: NON-EXISTING/FAKE DNS HOSTS in Via, Contact, Record-Route, RURI

HOW: counting on non IP hosts from Via and Contact



capture cookbook

DDOS ATTACKS:

KAMAILIOWORLD 2013

DNS ATTACK EXAMPLE

TYPE: NON-EXISTING/FAKE DNS HOSTS in Via, Contact, Record-Route
HOW: counting on non IP hosts from Via and Contact

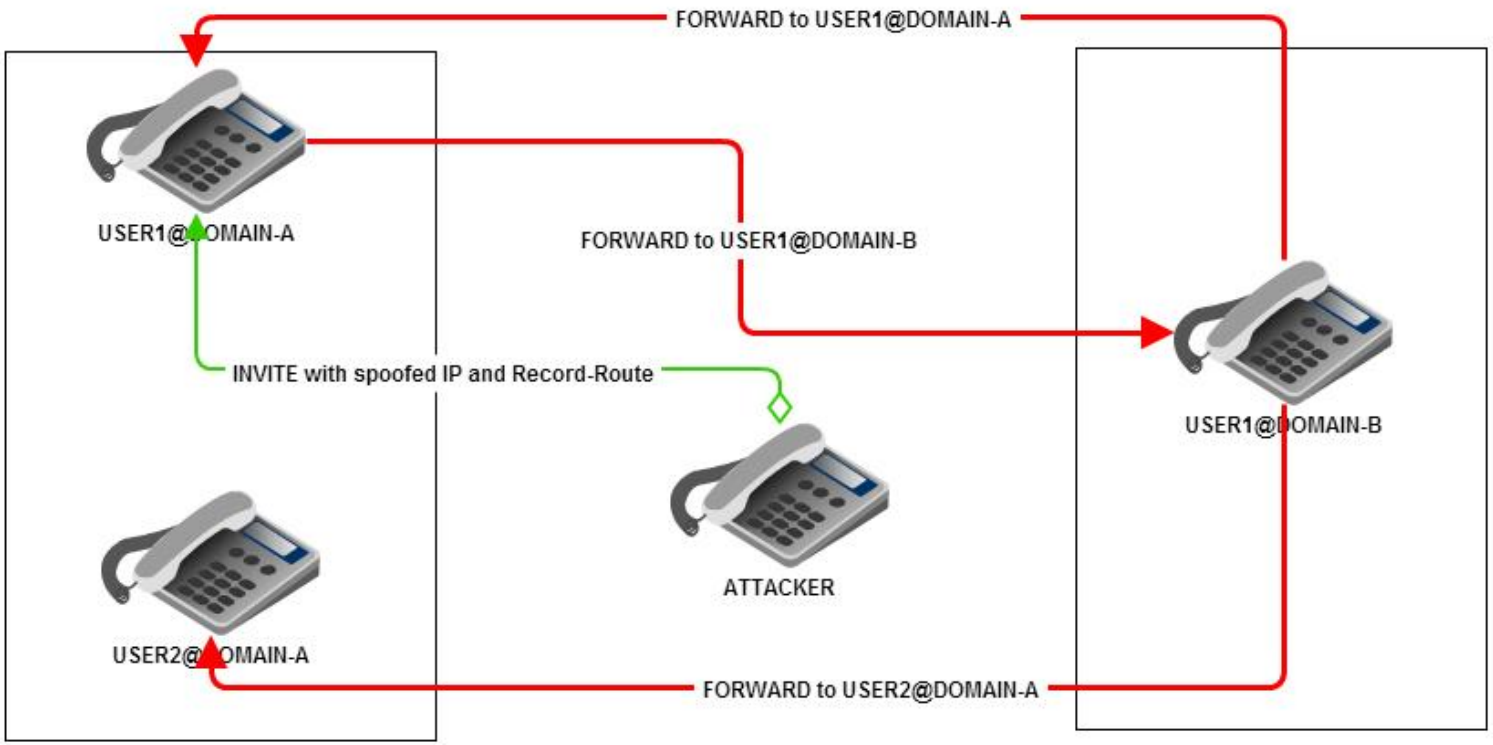
```
route {
...
        #Sample for IPv4
        if($sel(contact.uri.host) !~ "^(\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3})$") {
                if($sht(a=>alarm::dns) == $null) $sht(a=>alarm::dns) = 0;
                $sht(a=>alarm::dns) = $sht(a=>alarm::dns) + 1;
        }
...
        if($sel(via[1].host) !~ "^(\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3})$") {
                if($sht(a=>alarm::dns) == $null) $sht(a=>alarm::dns) = 0;
                $sht(a=>alarm::dns) = $sht(a=>alarm::dns) + 1;
        }
}
```

capture cookbook

DDOS ATTACKS:

HOMER
KAMAILIOWORLD 2013

AMPLIFICATION EXAMPLE DIAGRAM:



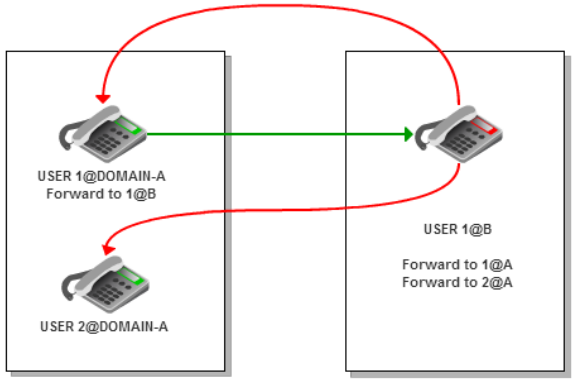
CAPTURE COOKBOOK

DDOS ATTACKS:

KAMAILIOWORLD 2013

AMPLIFICATION EXAMPLE:

TYPE: Message routing spoofing
HOW: counting on paar $fU $rU and on INVITES / 480




```
route {
        ...
        if($sht(a=>amplification::$fU::$rU) == $null)  $sht(a=>amplification::$fU::$rU) = 0;
        $sht(a=>amplification::$fU::$rU) = $sht(a=>amplification::$fU::$rU) + 1;
          ....
         if (is_method("INVITE")) {
                if($sht(a=>method::invite) == $null) $sht(a=>method::invite) = 0;
                $sht(a=>method::invite) = $sht(a=>method::invite) + 1;
         }
        ...
}
```

capture cookbook

APPLICATION LAYER ATTACKS:

KAMAILIOWORLD 2013

SIP SQL INJECTION

TYPE: SQL QUERIES IN SOME VARIABLES WHICH SHOULD BE GO THROUGH DB,
HOW: checking SQL patterns in some internal variables. i.e. authuser ($au)

EXAMPLE: auth user has SQL update:

```
Authorization: Digest username="2141; UPDATE subscriber SET password = '12345' WHERE username='admin'--",
realm="sip.provider.com", nonce="83b2c7fc-a59e-11e2-866e-f9ff32dafea1", uri="sip:sip.provider.com",
response="18caf84ee7105cecfcec92447b759aaa", algorithm=MD5, cnonce="ED8E67FE5614EAE7", qop=auth,
nc=00001310
```

```
route {
        .....

        if($au =~ "(\=)|(\-\-)|(\')|(\#)|(\%27)|(\%24)") {
                if($sht(a=>alarm::sqlinjection) == $null) $sht(a=>alarm::sqlinjection) = 0;
                 $sht(a=>alarm::sqlinjection) = $sht(a=>alarm::sqlijnection) + 1;
          }

        ....
}
```

capture cookbook

APPLICATION LAYER ATTACKS:

KAMAILIOWORLD 2013

PARSER ATTACKS

TYPE: bad packets in SIP and SDP trying to probe the parser/stack
HOW: counting on parser errors

413 Request Entity Too Large, **513 Too Large** and **400 Bad Request**


```
onreply_route {
        ....
         if(status == "413") {
                if($sht(a=>alarm::413) == $null) $sht(a=>alarm::413) = 0;
                $sht(a=>alarm::413) = $sht(a=>alarm::413) + 1;
          }

        ....

         # 400
         else if(status == "400") {

                if($sht(a=>alarm::400) == $null) $sht(a=>alarm::400) = 0;
                $sht(a=>alarm::400) = $sht(a=>alarm::400) + 1;

         }
}
```

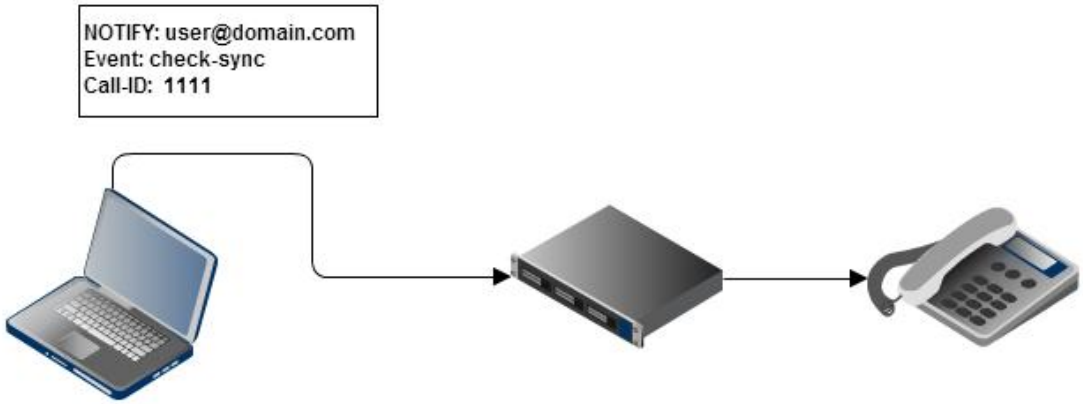
capture cookbook

APPLICATION LAYER ATTACKS:

KAMAILIOWORLD 2013

REMOTE MANIPULATION to SIP Phones w/ special features activated through Events and special headers

TYPE: send INVITE to target with header Call-Info: answer-after=0 and we can listen to what's going on in the room :-) SPY listening...



capture cookbook

APPLICATION LAYER ATTACKS:

HOMER
KAMAILIOWORLD 2013

REMOTE MANIPULATION to SIP phone through Events and special headers

TYPE: **NOTIFY** message with custom Event can reboot the phone. Using answer-after, can give possibility listen B-Party without confirmation.
HOW: counting on **NOTIFY** with "**Event: check-sync**" or **INVITE** with "**Call-Info: answer-after=0**"

```
route {
        ....
        if(method == "NOTIFY" && is_present_hf("Event") && $hdr(Event) == "check-sync" )
        {
                if($sht(a=>alarm::event::reboot) == $null) $sht(a=>alarm::event::reboot) = 0;
                $sht(a=>alarm::event::reboot) = $sht(a=>alarm::event::reboot) + 1;
        }
        ....

        if(method == "INVITE" && is_present_hf("Call-Info") && $hdr(Call-Info) =~ "answer-after" )
        {
                if($sht(a=>alarm::event::aa) == $null) $sht(a=>alarm::event::aa) = 0;
                $sht(a=>alarm::event::aa) = $sht(a=>alarm::event::aa) + 1;
        }
        ...
}
```

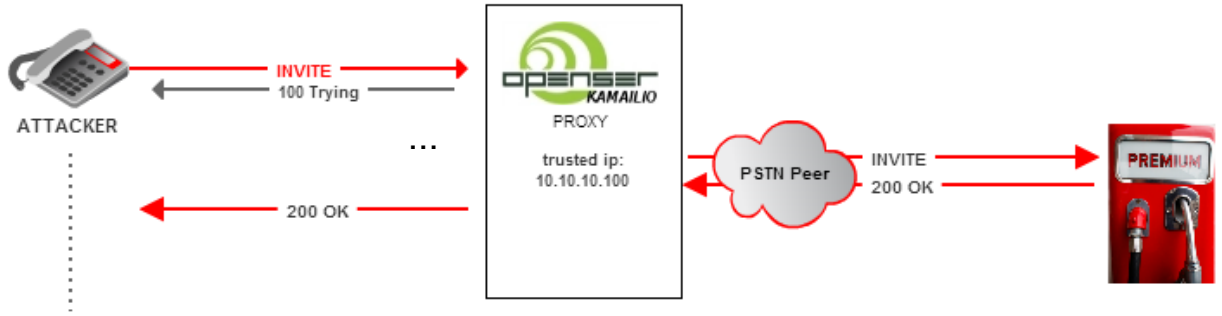
CAPTURE COOKBOOK

APPLICATION LAYER ATTACKS:

KAMAILIOWORLD 2013

SPOOFING WITH CUSTOM RECORD-ROUTE OR VIA

TYPE: Trusted Source IP spoofing . INVITE has IP adress of hacker in SIP headers(Record-Route, Via)
HOW: checking source_ip and IP from (Via,Record-Route) in case not NAT connection




```
route {
        ...
                if($sel(via[1].host) != $si) {
                        if($sht(a=>alarm::spoofing) == $null) $sht(a=>alarm::spoofing) = 0;
                        $sht(a=>alarm::spoofing) = $sht(a=>alarm::spoofing) + 1;
                }
            ...
                if($(hdr(Record-Route)[0]{nameaddr.uri}) != $si) {
                        if($sht(a=>alarm::spoofing) == $null) $sht(a=>alarm::spoofing) = 0;
                        $sht(a=>alarm::spoofing) = $sht(a=>alarm::spoofing) + 1;
                }
            ....
}
```

capture cookbook

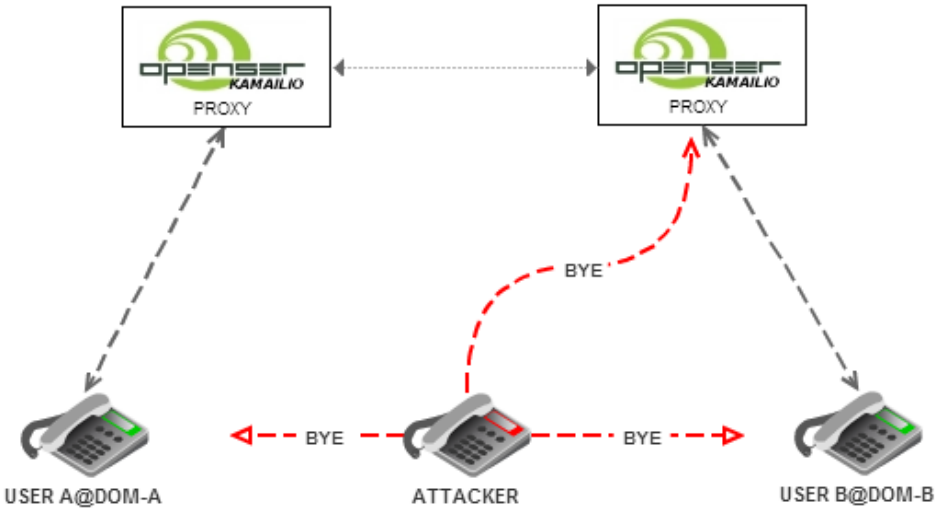
KAMAILIOWORLD 2013

HIJACKING:

SESSION TEARDOWN ATTACKS

TYPE: Session termination from third party

HOW: checking BYE IP and checking SIP Response code
481 (call transaction doesn't exist) for BYE




```
route {
        ....
            if($si != $sht(a=>ipinit::aleg::$ci) && $si != $sht(a=>ipinit::bleg::$ci)) {
                if($sht(a=>alarm::sessiontd) == $null) $sht(a=>alarm::sessiontd) = 0;
                    $sht(a=>alarm::sessiontd) = $sht(a=>alarm::sessiontd) + 1;
             }
             ....
}
onreply_route {
            if($rm == "BYE"  && status == "481") {
                    if($sht(a=>alarm::sessiontd) == $null) $sht(a=>alarm::sessiontd) = 0;
                     $sht(a=>alarm::sessiontd) = $sht(a=>alarm::sessiontd) + 1;
            }
}
```

capture cookbook

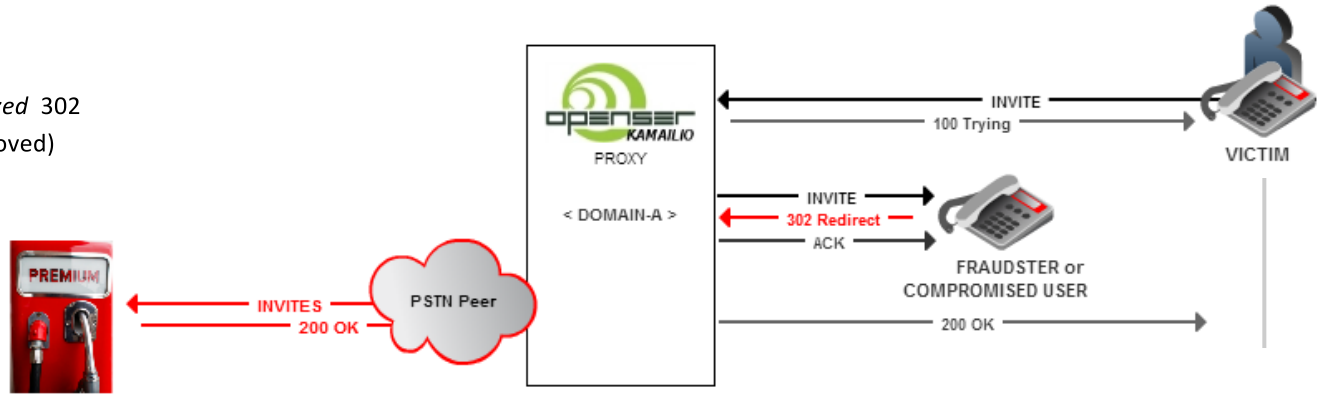
KAMAILIOWORLD 2013

HIJACKING:

REDIRECTION CALL HIJACKING

TYPE: Call hijacking by *unauthorized* 302

HOW: counting on 301 or 302 (Moved)




```
onreply_route {
        ....

          # MOVED
         if(status =~ "^(30[12])$") {
                if($sht(a=>response::301) == $null) $sht(a=>response::301) = 0;
                $sht(a=>response::301) = $sht(a=>response::301) + 1;
           }
        ....
}
```

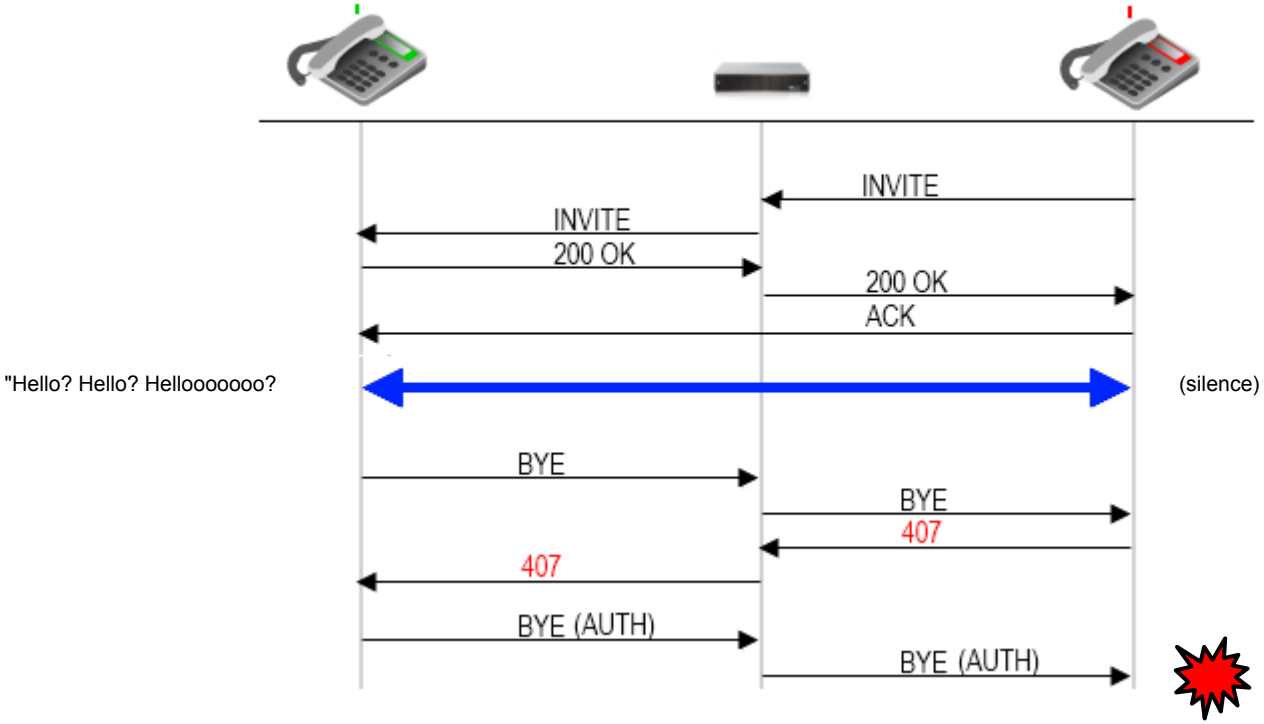
capture cookbook

KAMAILIOWORLD 2013

HIJACKING:

PASSWORD HIJACKING ATTEMPTS (also doubles as Auth DB issues detection/premonition on early signs of collapse)

TYPE: brute force for INVITE or auth response on BYE



capture cookbook

HIJACKING:

PASSWORD HIJACKING ATTEMPTS (also doubles as Auth DB issues detection/premonition on early signs of collapse)

TYPE: brute force for INVITE or auth response on BYE

HOW: counting on 401/407 for INVITE /REGISTER (*BRUTE FORCE*) and BYE (401/407)

```
onreply_route {
        ....
        if($rm == "INVITE") {
                if(status == "407") {
                    if($sht(a=>response::407::invite) == $null) $sht(a=>response::407::invite)= 0;
                    $sht(a=>response::407::invite) = $sht(a=>response::407::invite) + 1;
                }
                else if(status == "401") {
                if($sht(a=>response::401::invite) == $null) $sht(a=>response::401::invite)= 0;
                    $sht(a=>response::401::invite) = $sht(a=>response::401::invite) + 1;
                }
        }
        else if($rm == "BYE") {
                if(status == "407") {
                    if($sht(a=>response::407::bye) == $null) $sht(a=>response::407::bye)= 0;
                    $sht(a=>response::407::bye) = $sht(a=>response::407::bye) + 1;
                }
                else if(status == "401") {
                if($sht(a=>response::401::bye) == $null) $sht(a=>response::401::bye)= 0;
                    $sht(a=>response::401::bye) = $sht(a=>response::401::bye) + 1;
                }
        }
}
```

capture cookbook

FRAUD/SCAMMING:

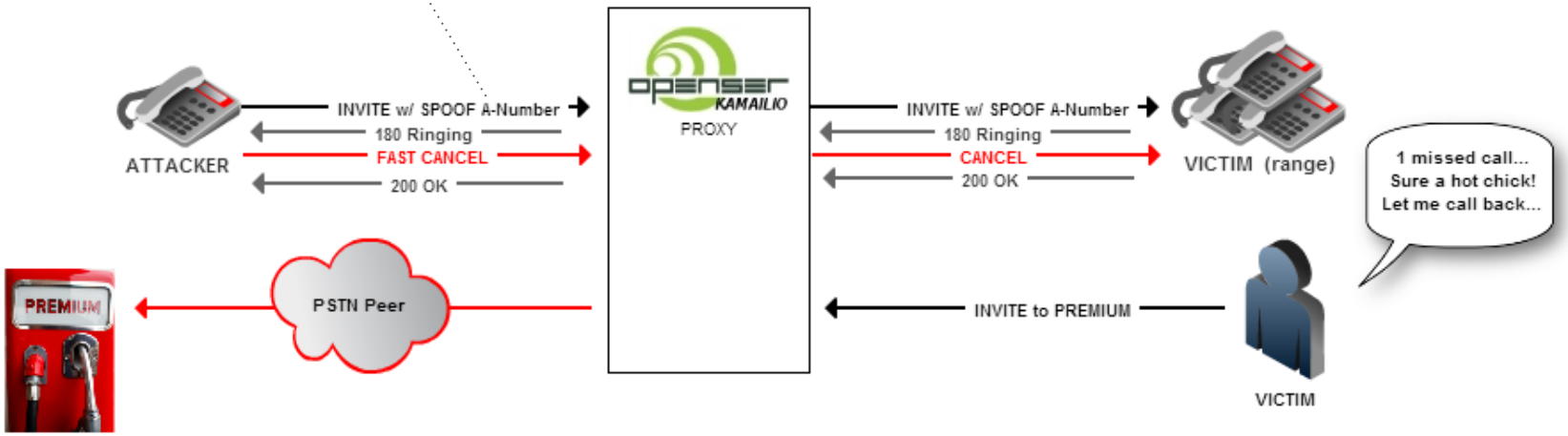
KAMAILIOWORLD 2013

CALLBACK FRAUD

INVITE sip:1777@sip.provider.com SIP/2.0
From: "+49900111111" <sip:021444444@sip.provider.com>;
tag=befeeead3e
...

OR in case CLIP NO SCREENING:

INVITE sip:1777@sip.provider.com SIP/2.0
From: <sip:+49900111111@sip.provider.com>;tag=befeeead3e
P-Asserted-Identity: <sip:021444444@192.168.0.201>
....



CAPTURE COOKBOOK

FRAUD/SCAMMING:

KAMAILIOWORLD 2013

CALLBACK FRAUD

TYPE: Caller spoofs a Premium Number and just make a short call attempts without connect. Calling back the number usually produces high bills by playing back ring signal on an already established call to gain minutes.

HOW: checking A-Number again known list of banned A-number prefixes/numbers

```
route {
        ....
        $var(anumber) = $fU; #Here can be check for Diversion, History etc

        if($var(anumber)=~ "^+49900$") {
                if($sht(a=>alarm::scam) == $null) $sht(a=>alarm::scam) = 0;
                 $sht(a=>alarm::scam) = $sht(a=>alarm::scam) + 1;
        }
              ....
        # OR we can use central DB for all numbers
        sql_query("ca", "SELECT * FROM scam_codes WHERE code = '$var(anumber)'", "ra");
        if($dbr(ra=>rows)>0)
        {
                if($sht(a=>alarm::scam) == $null) $sht(a=>alarm::scam) = 0;
                 $sht(a=>alarm::scam) = $sht(a=>alarm::scam) + 1;
        }
        sql_result_free("ra");
        ...
}
```

ALARMS

& NOTIFICATIONS

KAMAILIOWORLD 2013

Great - We now have new alarms!

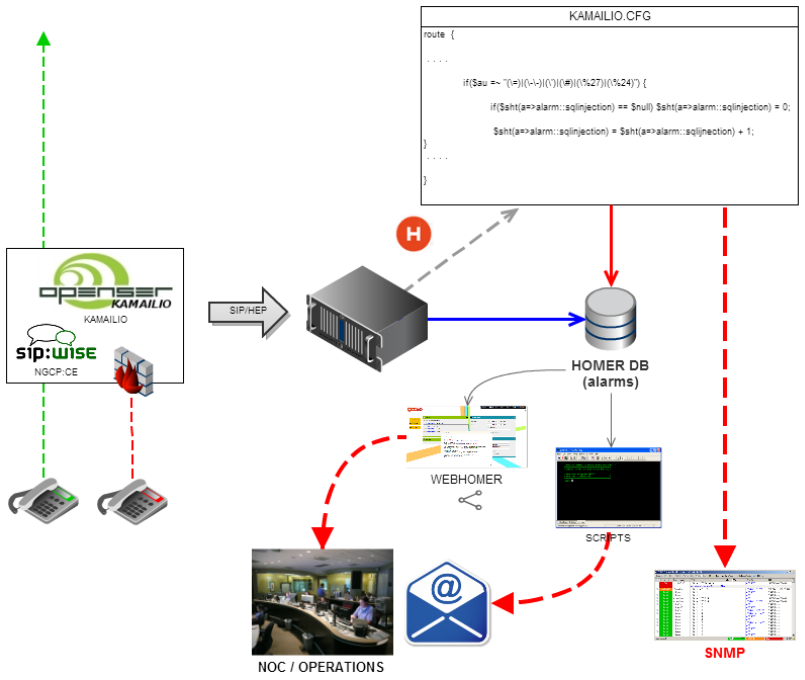
How can we inform the NOC ?

If alarms exceed safe capacity limitations (i.e. 1000 INVITES p/s) there are several ways to inform NOC:

- WebHomer or Homer API Scripting
  (timer module and execute route block)
- DB monitoring scripts
- Send SNMP traps / EMAIL from Kamailio
- Robocall to NOC and play notifications

Alternative?

- Wait for the HUGE interconnection bills
- Wait for Customers to cancel service
- Wait for Equipment to melt down
- Spend 250k+ for proprietary solutions



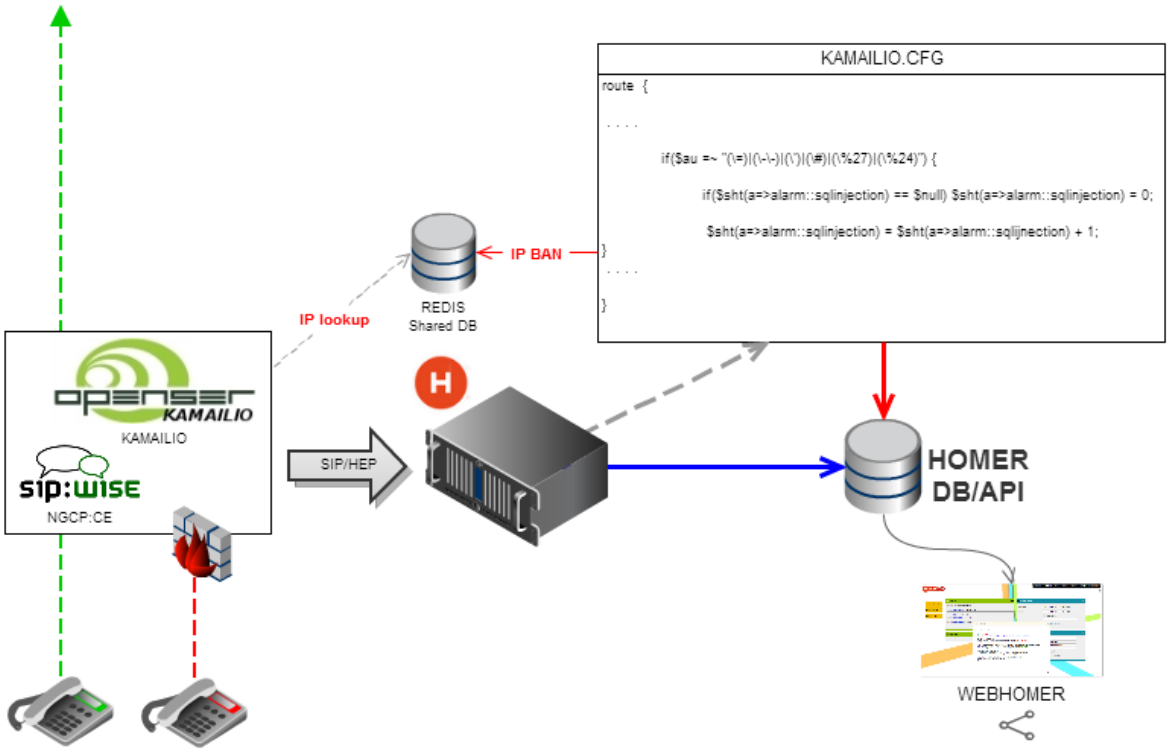
Safety & Prevention

Proactive examples

KAMAILIOWORLD 2013

My NOC is too slow! **How can we automatically block the offending source/destinations during investigation?**

If you have a big cluster in your network, use a centralized DB or REDIS system storing alarm reported IPs/Routes. Each proxy node should check global ***blacklist*** and on a match, reject/drop the session and source until it's cleared.



Session Quality

and other useful detections

KAMAILIOWORLD 2013

How to detect other attacks and bad session quality?

- Generate statistics on important methods/replies
- Calculate *ASR/NER/ISA/SD/SSR* variables

  SD = *Session Defects*
  [SUM(500,503,504)]

  ISA = *Ineffective Session Attempts*
  [SUM(408,500,503)]

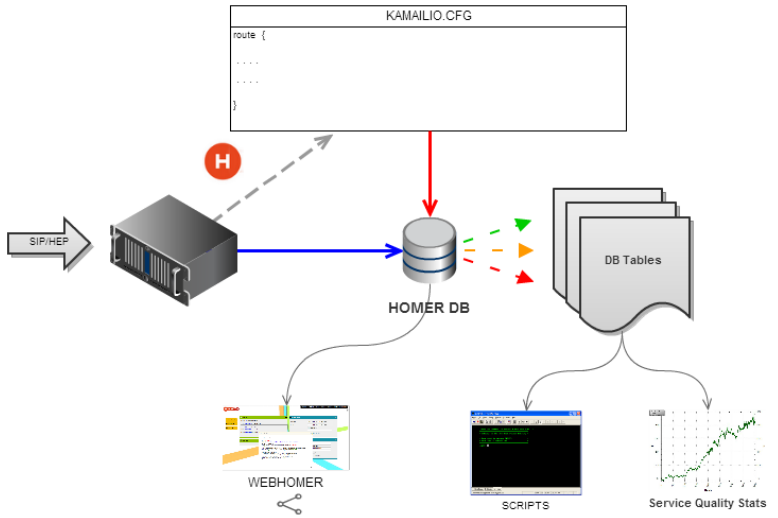
  AHR = *Average HOP Requests*

  ASR = *Answer Seizure Ratio*
  [('200' / (INVITES - AUTH - SUM(3XX))) * 100]

  NER = *Network Efficiency Ratio*
  [('200' + ('486','487','603') / (INVITES -AUTH-(SUM(30x)) * 100]

- Check cause codes in BYE *(Reason header)*

  If not 16 OR 17 then increase the relative counter



Feeling great usage ideas popping up?
Noticed something we did not cover?

Please come share on our Wiki!

http://homer.googlecode.com

HOMER / SIPCAPTURE

PROJECT UPDATES & ROADMAP

KAMAILIOWORLD 2013

Project Updates

Homer install base is growing - fast

Despite being maintained and developed by a relatively small core team, HOMER is already responsible for billions of SIP packets captured and searched each day worldwide

More and more are choosing HOMER over other solutions, too expensive or just not flexible

Small IP Telcos (OSS-aware) already consider HOMER a must-have tool for their operations

Big Telcos & Vendors are starting to show interest - we can't name no names, but they are!

Project Roadmap 2013

Q1 - Homer 3.5 developers release *(want to join us? support@sipcapture.org)*

Q2 - Homer 3.5 public release & Documentation

Q3 - Captagent 4.1 release (more features, more performance... more bug fixes :-)

Q4 - Next-Generation of the HOMER project will be revealed

EOF

NOT FOUND

KAMAILIOWORLD 2013

That's All Folks!

"HEP Yourself with Homer!"

Get in touch with us:

SUPPORT@SIPCAPTURE.ORG