Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Steffen Haas
Department of Computer Science
IT Security and Security Management (ISS)

Bro-Osquery

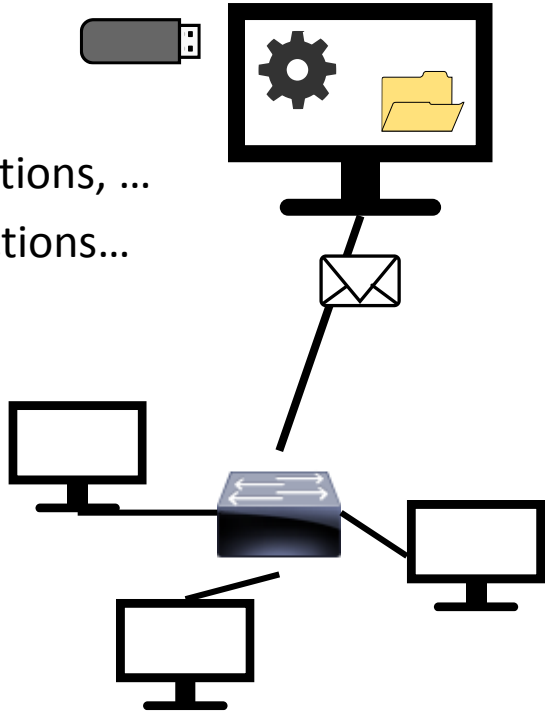
Bro Network Monitor
https://www.bro.org

Osquery Host Monitor
https://osquery.io/

Large-Scale Host and Network Monitoring Using Open-Source Software

Motivation

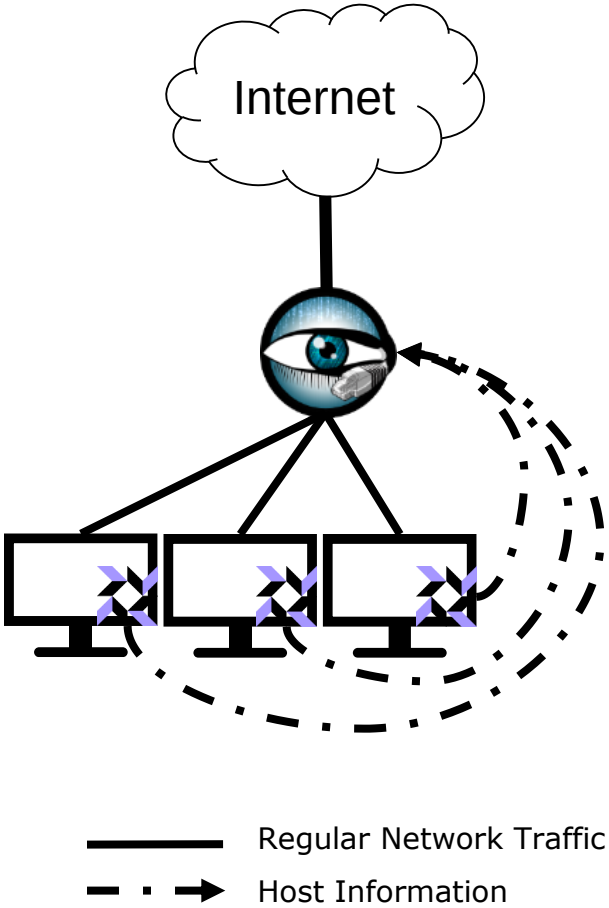
- Today: Separate monitoring of hosts and network
  - Sometimes not event both in place
  - Not enough visibility to detect all attacks
- Limited visibility in separated monitoring through
  - Hosts: Unknown or hiding malware, negligible local actions, …
  - Network: Encrypted network traffic, malicious local actions…
- Required: Assessing host behavior by evaluating
  - Activity on hosts (host-centric)
  - Communication with other hosts (network-centric)
  - Both at the same time!
- Benefits from combining host and network monitoring
  - More context about network communications
  - More context about communicating applications



Bro-Osquery in a nutshell

- Two types of data sources in your network
  - Network Monitor: Bro
  - Host Monitor: Osquery

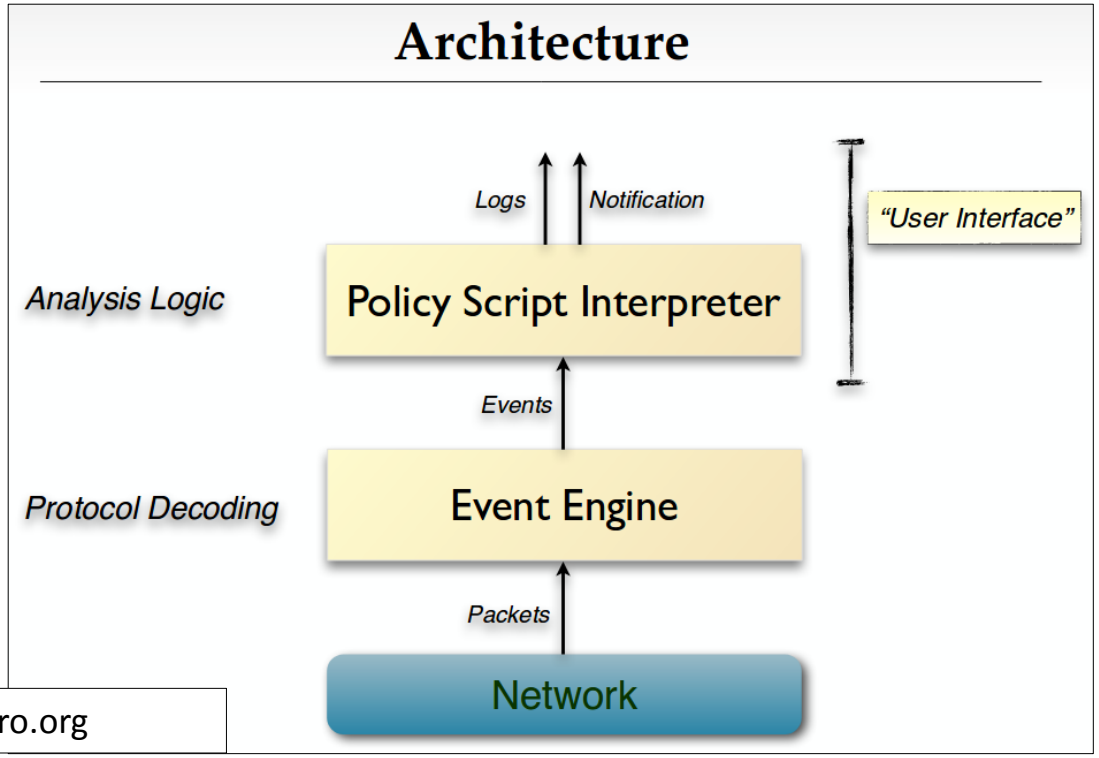
- Bro as central analysis platform
  - Monitors network communication
  - Receives data from Osquery hosts
  - Enables correlation of host and network data
    - Which app/user is responsible for specific communication?
  - Detection of (attack) scenarios with knowledge from hosts and network
    - Tracking execution of downloaded files
    - Detecting SSH-Chain
    - Identifying users responsible for data exfiltration



Bro in a nutshell

- Powerful analysis framework for network traffic
- Focuses on network security monitoring

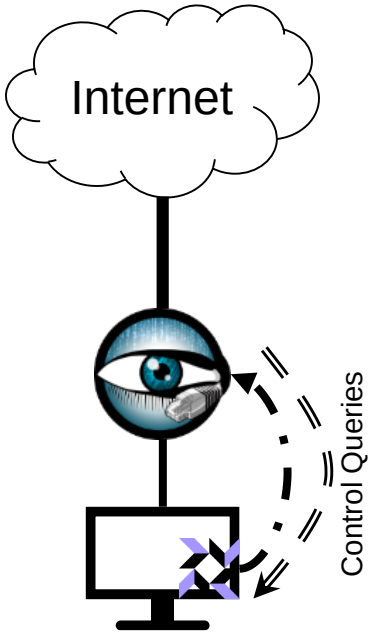
- Scripts as Bro's "Magic Ingredient"
  - Comes with > 10,000 lines of script code
  - Deep Packet Inspection
    - Application specific
  - Custom policies and detection rules



From https://www.bro.org

Features of Bro-Osquery

- Controlling Osquery schedule and receiving results with Bro
  - Central control instance for querying groups of Osquery hosts
    - Maintaining query schedule of hosts at runtime
    - Ability to execute one-time queries
  - Results are natively fed back and are available in Bro script



- Logging query results
  - Central logging of structured data as Bro log files
  - Extending network sessions with users/applications

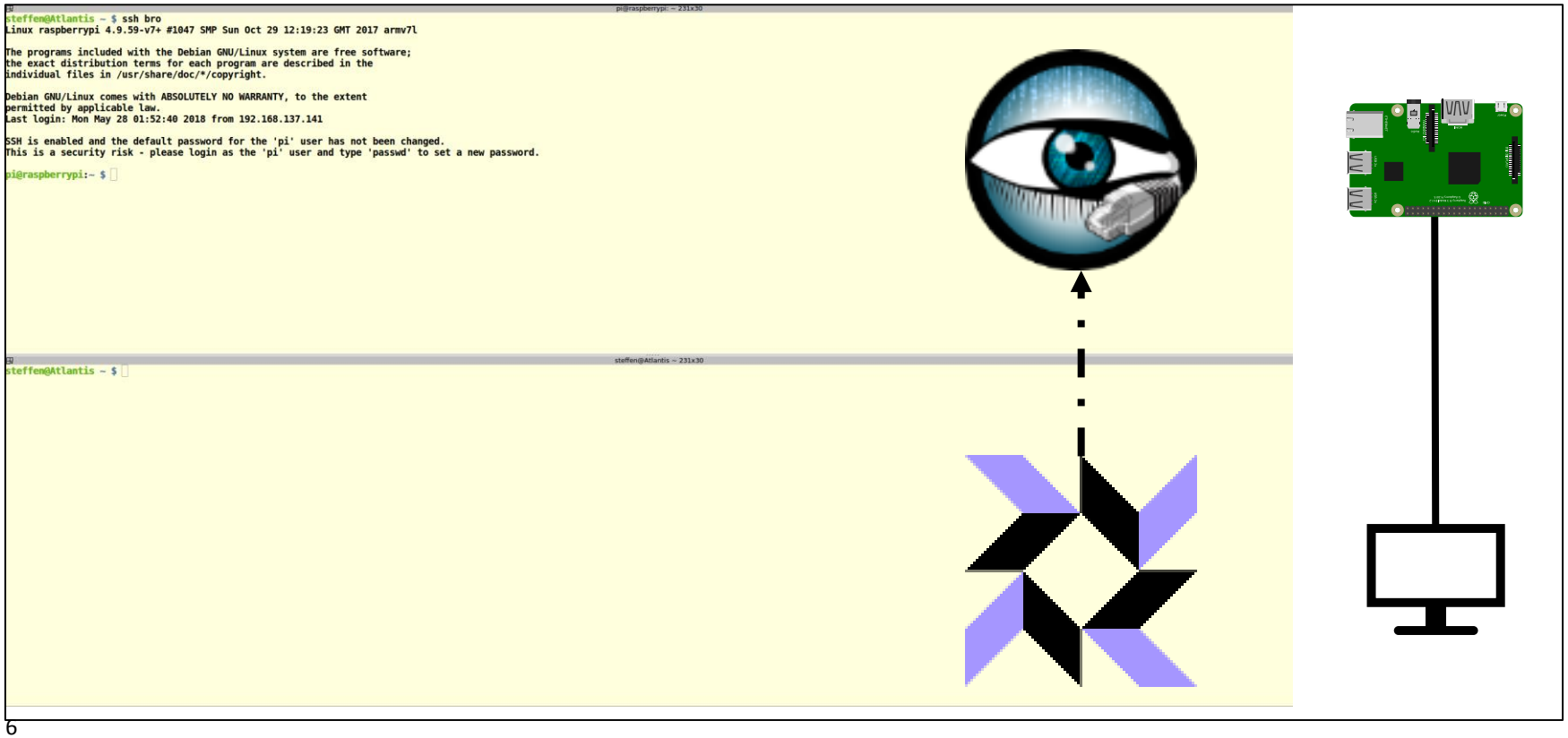
- Detection of sophisticated scenarios
  - Ability to write Bro scripts with access to full host and network data
  - Event-based detection in real-time extensible by custom scripts

- Large-scale deployments
  - Load distribution using proxies and/or multiple Bros

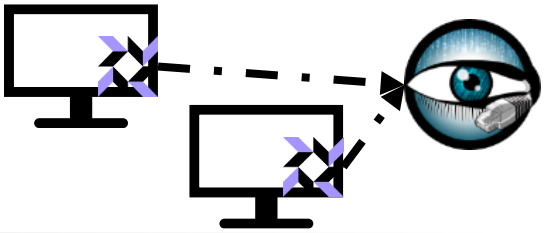
Demo: Logging of SQL Queries

- Controlling and logging the query results for all connected Osquery hosts

```
event bro_init()
{
  Log::create_stream(LOG, [$columns=Info, $path="osq-processes"]);

  local query = [$ev=host_processes,
    $query="SELECT pid,name,path,cmdline,cwd,root,uid,gid,on_disk,start_time,parent,pgroup FROM processes"];
  osquery::subscribe(query);
}
```



Demo: Logging of SQL Queries



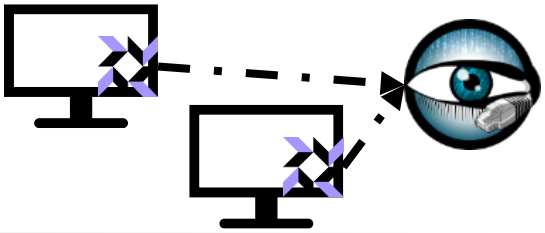
pi@raspberrypi: ~ 190x25

```
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun May 27 16:08:59 2018 from 192.168.137.141

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set a new password.

pi@raspberrypi:~ $ sudo /usr/local/bro/bin/broctl deploy
checking configurations ...
installing ...
removing old policies in /usr/local/bro/spool/installed-scripts-do-not-touch/site ...
removing old policies in /usr/local/bro/spool/installed-scripts-do-not-touch/auto ...
creating policy directories ...
installing site policies ...
generating standalone-layout.bro ...
generating local-networks.bro ...
generating broctl-config.bro ...
generating broctl-config.sh ...
stopping ...
stopping bro ...
starting ...
starting bro ...
pi@raspberrypi:~ $
```

steffen@Atlantis ~ 190x26

```
steffen@Atlantis ~ $ sudo rm -rf /var/osquery/osquery.db/ && sudo systemctl start osquery
steffen@Atlantis ~ $
```

Demo: Logging of SQL Queries



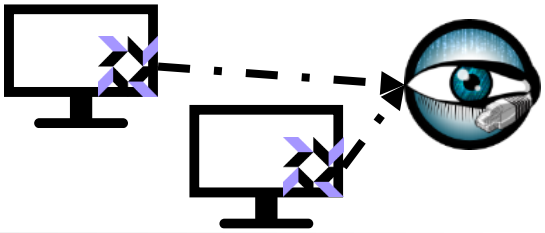
pi@raspberrypi: ~ 190x25

```
permitted by applicable law.
Last login: Sun May 27 16:08:59 2018 from 192.168.137.141

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set a new password.

pi@raspberrypi:~ $ sudo /usr/local/bro/bin/broctl deploy
checking configurations ...
installing ...
removing old policies in /usr/local/bro/spool/installed-scripts-do-not-touch/site ...
removing old policies in /usr/local/bro/spool/installed-scripts-do-not-touch/auto ...
creating policy directories ...
installing site policies ...
generating standalone-layout.bro ...
generating local-networks.bro ...
generating broctl-config.bro ...
generating broctl-config.sh ...
stopping ...
stopping bro ...
starting ...
starting bro ...
pi@raspberrypi:~ $ tail -n0 -f /usr/local/bro/logs/current/osq-processes.log
1527437698.622382       4C4C4544-0038-4710-8037-C2C04F504332    29958   nc      /bin/nc.openbsd nc google.de 80 /home/steffen   /       1000    1000    1       197522  4697    29958
^C
pi@raspberrypi:~ $
```

steffen@Atlantis ~ 190x26

```
steffen@Atlantis ~ $ sudo rm -rf /var/osquery/osquery.db/ && sudo systemctl start osquery
steffen@Atlantis ~ $ nc google.de 80
^C
steffen@Atlantis ~ $
```

Demo: Logging of SQL Queries



pi@raspberrypi: ~ 190x25

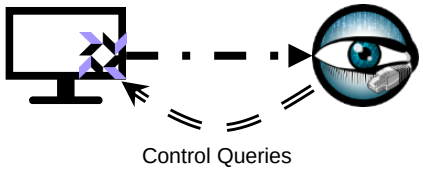
```
pi@raspberrypi:~ $ sudo /usr/local/bro/bin/broctl deploy
checking configurations ...
installing ...
removing old policies in /usr/local/bro/spool/installed-scripts-do-not-touch/site ...
removing old policies in /usr/local/bro/spool/installed-scripts-do-not-touch/auto ...
creating policy directories ...
installing site policies ...
generating standalone-layout.bro ...
generating local-networks.bro ...
generating broctl-config.bro ...
generating broctl-config.sh ...
stopping ...
stopping bro ...
starting ...
starting bro ...
pi@raspberrypi:~ $ tail -n0 -f /usr/local/bro/logs/current/osq-processes.log
1527437698.622382       4C4C4544-0038-4710-8037-C2C04F504332    29958   nc      /bin/nc.openbsd nc google.de 80 /home/steffen   /       1000    1000    1       197522  4697    29958
^C
pi@raspberrypi:~ $ tail -n0 -f /usr/local/bro/logs/current/osq-process_events.log
1527437810.678441       4C4C4544-0038-4710-8037-C2C04F504332    30576   /bin/bash       "bash"  (empty) 1000    1000    1527437809      4697
1527437810.678441       4C4C4544-0038-4710-8037-C2C04F504332    30576   /usr/bin/curl   curl google.de  /home/steffen   1000    1000    1527437809      4697
1527437810.678441       4C4C4544-0038-4710-8037-C2C04F504332    30577   /usr/bin/curl   "curl"  (empty) 1000    1000    1527437809      30576
^C
pi@raspberrypi:~ $
```

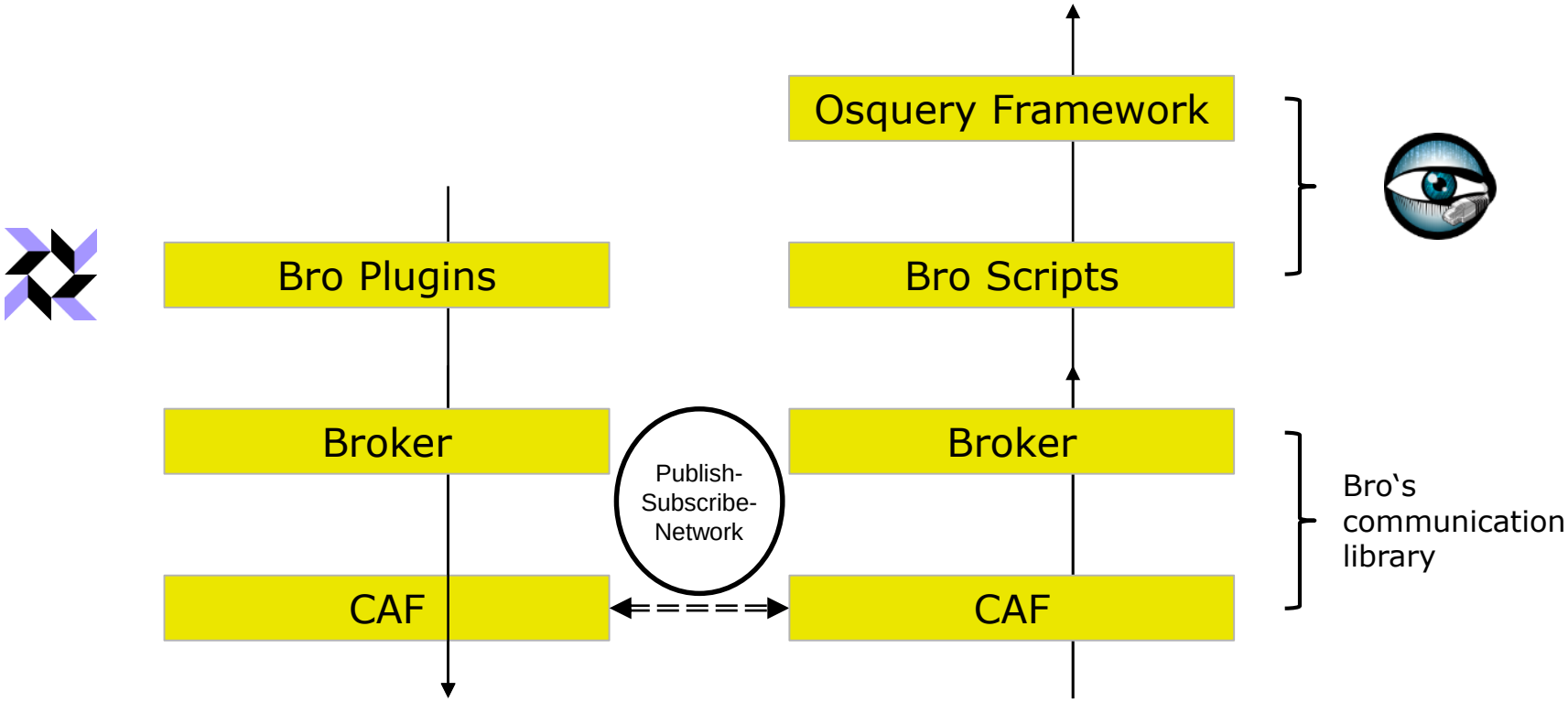
steffen@Atlantis ~ 190x26

```
steffen@Atlantis ~ $ sudo rm -rf /var/osquery/osquery.db/ && sudo systemctl start osquery
steffen@Atlantis ~ $ nc google.de 80
^C
steffen@Atlantis ~ $ curl google.de
<HTML><HEAD><meta http-equiv="content-type" content="text/html;charset=utf-8">
<TITLE>301 Moved</TITLE></HEAD><BODY>
<H1>301 Moved</H1>
The document has moved
<A HREF="http://www.google.de/">here</A>.
</BODY></HTML>
steffen@Atlantis ~ $
```

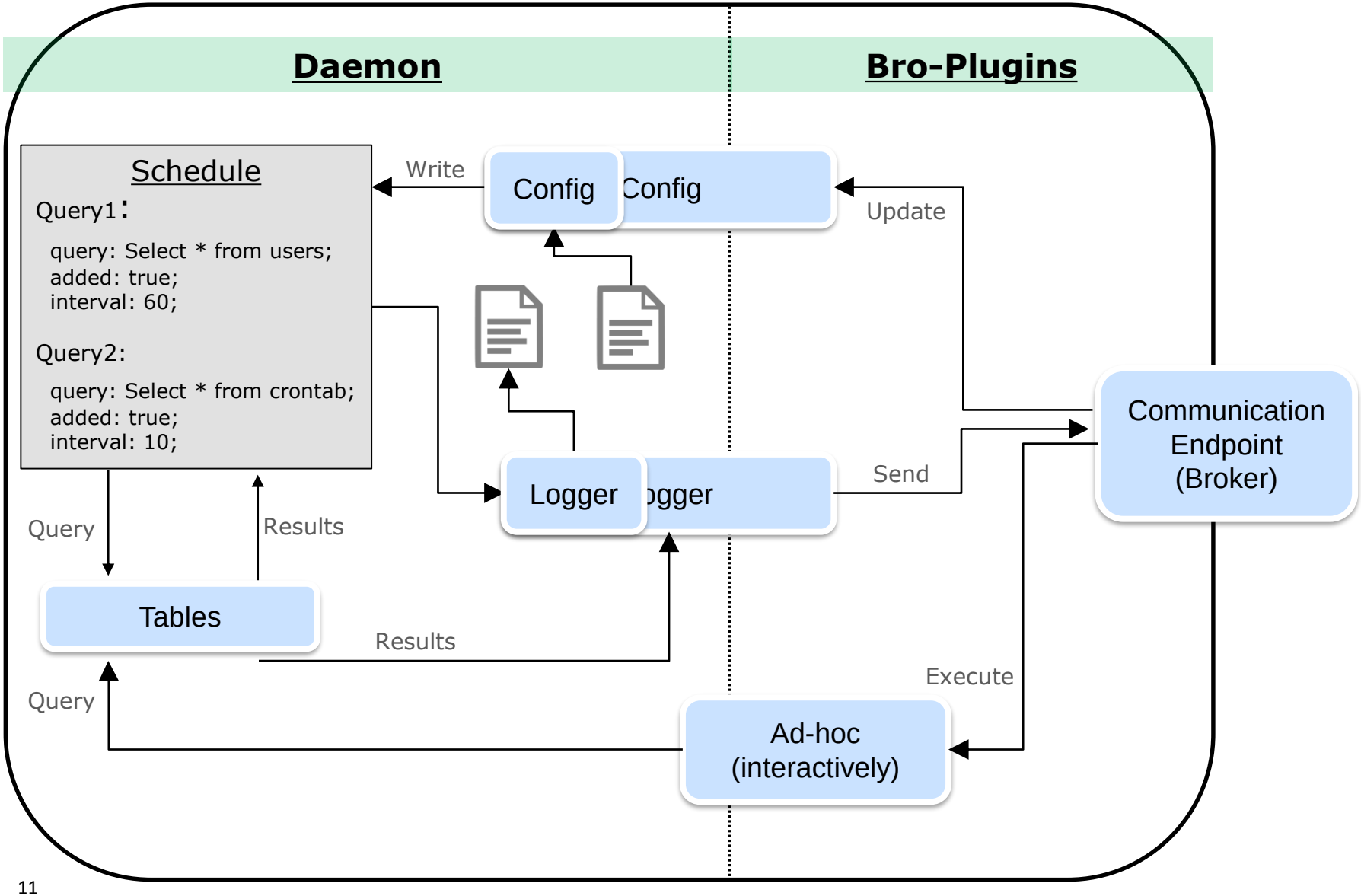
Network Stack in Bro-Osquery

- Extensions to the existing open-source tools
  - In Osquery:
    - Bro plugins including communication library (c++)
  - In Bro:
    - Osquery framework (bro script)





Architecture in Osquery



Technical Details: Extending Osquery Code

- Broker Manager (Singleton)
  - Connectivity with the Broker network
  - Handling of messages (publishing and subscribing to messages)

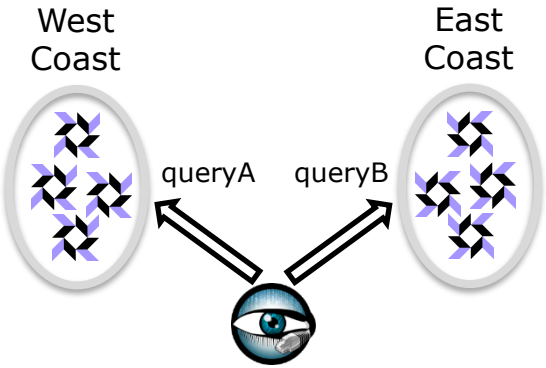
- Query Manager (Singleton)
  - State keeping of schedule/ad-hoc queries for result handling

- Distributed Plugin
  - “Runnable” to receive Broker messages
  - Updating schedule or execution of one-time queries

- Logger Plugin
  - Sending query results to Bro

Using the Osquery Framework

- Organization of Osquery hosts
  - Hosts are organized in groups (non-disjoint)
    - Statically by configuration
    - Dynamically based on IP subnets
  - Groups can be addressed by SQL queries
  - Default group contains all Osquery hosts



- Communication with Osquery hosts
  - API for organizing groups (IP subnet -> group name)

```
global set_host_group: function(range: subnet, group: string);
```

  - API for subscribing queries (query result -> topic name)

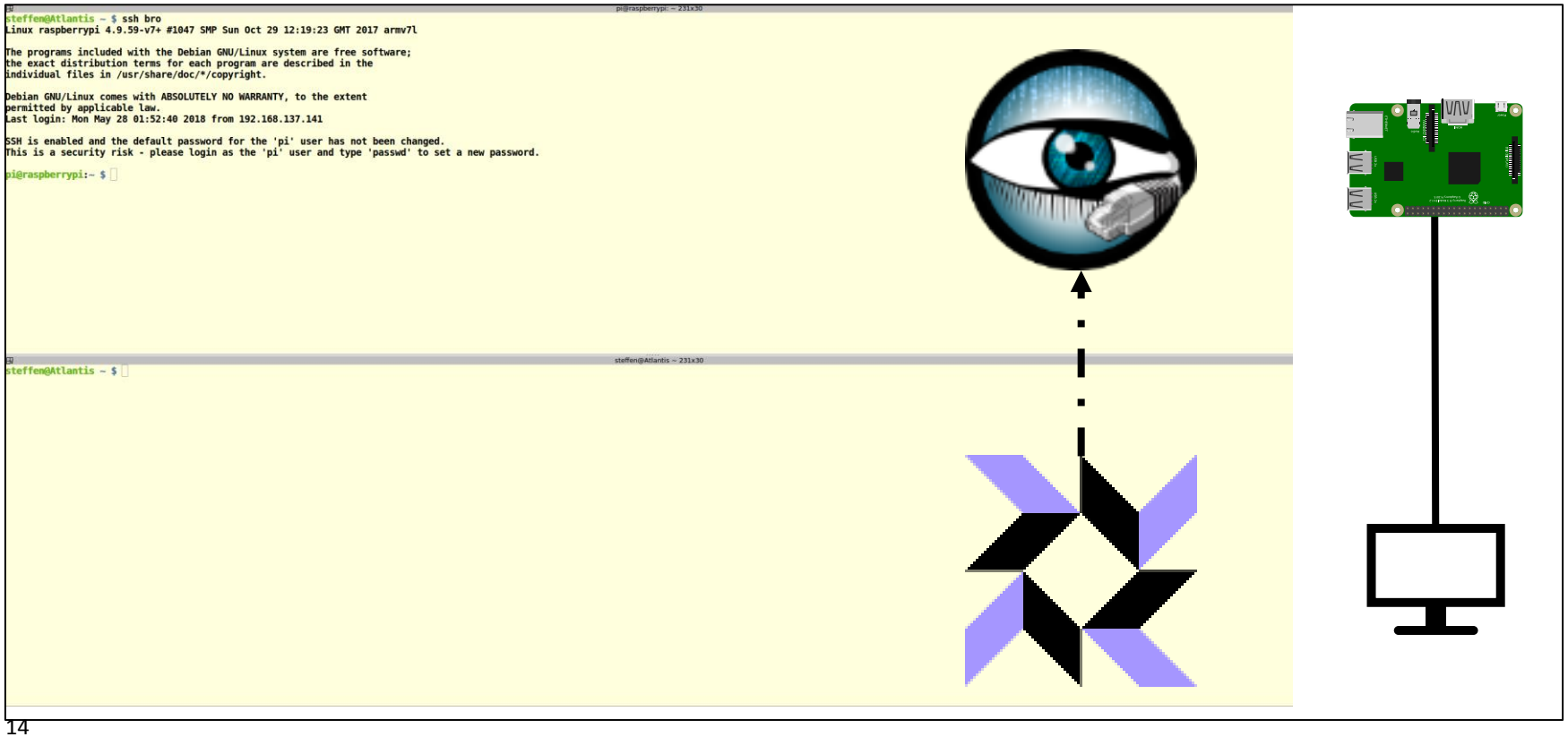
```
global subscribe: function(q: Query, host: string &default="", group: string &default="");
```

  - API for executing one-time queries (query result -> topic name)

```
global execute: function(q: Query, host: string &default="", group: string &default="");
```

Demo: Host-Network Correlation

- Tie username and process to TCP connections



Demo: Host-Network Correlation

pi@raspberrypi: ~ 190x25

```
pi@raspberrypi:~ $ tail -n0 -f /usr/local/bro/logs/current/osq-user_connections.log
1527439274.704343       4C4C4544-0038-4710-8037-C2C04F504332    connect 0.0.0.0 0       192.168.137.1   53      6520    /bin/nc.openbsd nc google.de 80 1000    steffen
1527439274.704343       4C4C4544-0038-4710-8037-C2C04F504332    connect 0.0.0.0 0       172.217.16.67   80      6520    /bin/nc.openbsd nc google.de 80 1000    steffen
1527439274.704343       4C4C4544-0038-4710-8037-C2C04F504332    connect 0.0.0.0 0       2a00:1450:4005:800::2003        80      6520    /bin/nc.openbsd nc google.de 80 1000    steffen
1527439274.704343       4C4C4544-0038-4710-8037-C2C04F504332    connect 0.0.0.0 0       172.217.16.67   80      6520    /bin/nc.openbsd nc google.de 80 1000    steffen
^C
pi@raspberrypi:~ $
```

steffen@Atlantis ~ 190x25

```
steffen@Atlantis ~ $ nc google.de 80
^C
steffen@Atlantis ~ $
```

Demo: Host-Network Correlation

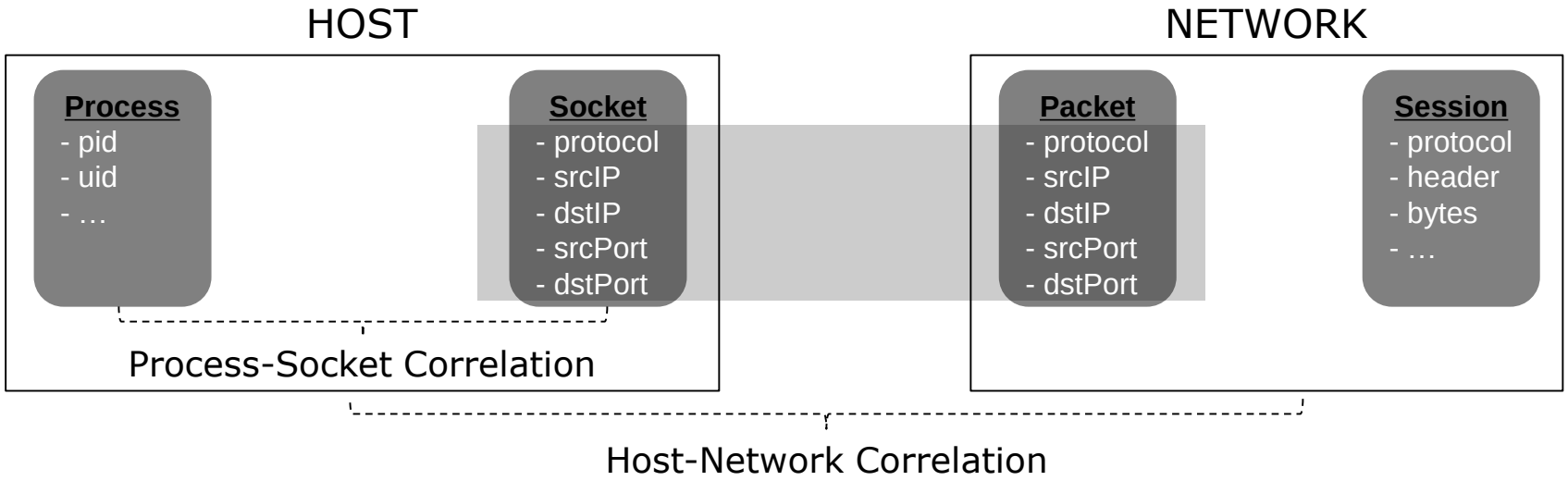
pi@raspberrypi: ~ 190x25

```
pi@raspberrypi:~ $ tail -n0 -f /usr/local/bro/logs/current/conn.log
1527439383.834317       Cc6H7n1ZFkX1BLVWu5      192.168.137.141 59710   172.217.16.67   80      tcp     http    0.059120        73      538     SF      T       F       0       ShADadFf        6
393     4       754     -       7151    /usr/bin/curl   steffen -       -       -
1527439383.831653       CYqr0O3r7c0wbIzaNh      192.168.137.141 54066   192.168.137.1   53      udp     dns     0.000002        54      0       S0      T       T       0       Dc      2       1
10      0       0       -       7152    /usr/bin/curl   steffen -       -       -
^C
pi@raspberrypi:~ $
```

steffen@Atlantis ~ 190x25

```
steffen@Atlantis ~ $ curl google.de
<HTML><HEAD><meta http-equiv="content-type" content="text/html;charset=utf-8">
<TITLE>301 Moved</TITLE></HEAD><BODY>
<H1>301 Moved</H1>
The document has moved
<A HREF="http://www.google.de/">here</A>.
</BODY></HTML>
steffen@Atlantis ~ $
```

Process-Socket Correlation

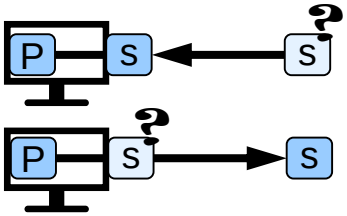


- Process-Socket Correlation based on audit
  - Processes: Event-based table "process_events"
  - Socket: Event-based table "socket_events"
    - Incomplete five-tuple socket
    - Two possible socket actions: "bind" and "connect"

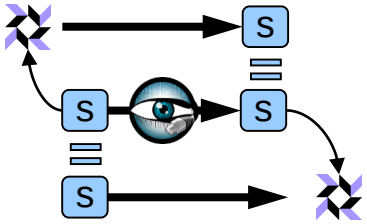
action	protocol	local_addr	local_port	remote_addr	remote_port
connect	✗	✗	✗	<remote_addr>	<remote_port>
bind	✗	<local_addr>	<local_port>	✗	✗

Host-Network Correlation

- Process-Socket Correlation
  - Merging of process/socket events based on common process ID
  - Process-Socket data of each host
    - Socket binds on local IPs and ports
    - Socket connects to remote IPs and ports



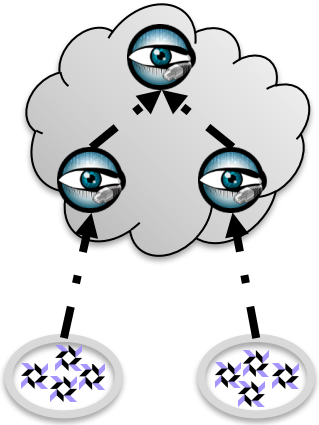
- Host-Network Correlation for specific network connection
  - Matching the five-tuples that identify
    - Sockets on hosts
    - Connections in the network



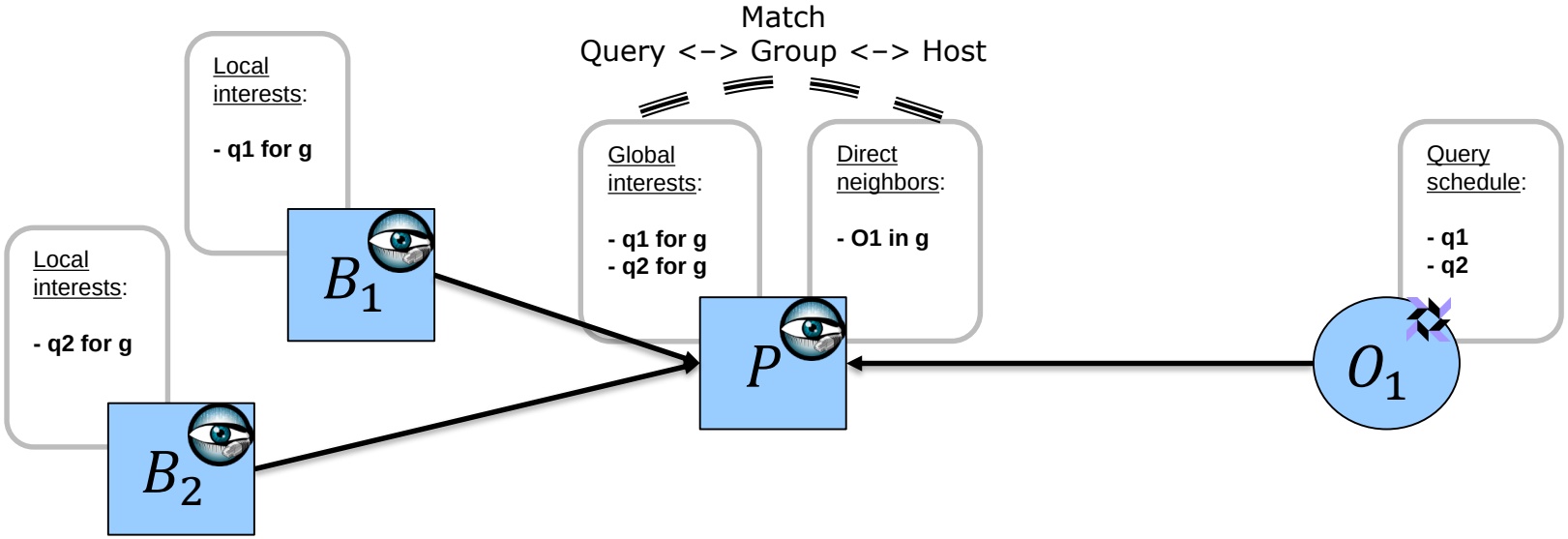
- Host-Network Correlation with Process-Socket Correlation based on audit
  - Identify hosts for source and destination IP of the connection
  - Search the Process-Socket data of the two hosts for specific network connection
    - Source host: Match remote address (IP+Port) only
    - Destination host: Match local address (IP+Port) only

Large-Scale Deployments

- Load distribution through proxies and multiple Bros
  - Backbone consists out of Bros and proxies
  - Queries of interest pushed to backbone edges
  - Osquery hosts connect to an edge Bro/proxy



- Distribution of interests



Project Status of Bro-Osquery

- Complete view on processes
  - Using event-based table to capture short-lived processes
  - Table contains only “execve” syscalls
  - Network communication probably by asynchronous threads
    - Created by “fork”/”clone” syscall

- Upgrade to osquery 3
  - Redesign of the kernel audit in Osquery 3
  - Breaks the event-based tables when Osquery schedule is updated
    - Although updating schedule is an external API (github issue)
  - Bro-Osquery is stuck on latest Osquery 2 (2.11.2 from Dec 30, 2017)

- Large-scale testbed
  - Are you interested in running Bro-Osquery?

How to run Bro-Osquery?

- Project repository:
  - https://github.com/bro/bro-osquery

- Install Bro-featured Osquery
  - Build from fork until Bro is officially supported
  - Optionally: Set up as service and write configuration file

- Install Osquery-featured Bro
  - Build from source for required development features
  - Install the osquery framework as Bro scripts
  - Use existing/custom Bro scripts to query Osquery hosts

Questions?