The Bro Network Security Monitor

Broverview

Outline

Outline

Philosophy and Architecture

A *framework* for network traffic analysis.

Outline

Philosophy and Architecture

A *framework* for network traffic analysis.

History

From research to operations.

Outline

Philosophy and Architecture

A *framework* for network traffic analysis.

History

From research to operations.

Architecture

Components, logs, scripts, cluster.

What is Bro?

What is Bro?

Packet Capture

What is Bro?

Packet Capture

Traffic Inspection

What is Bro?

Packet Capture

Traffic Inspection

Attack Detection

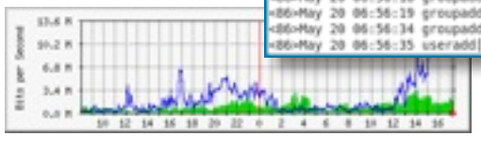
What is Bro?

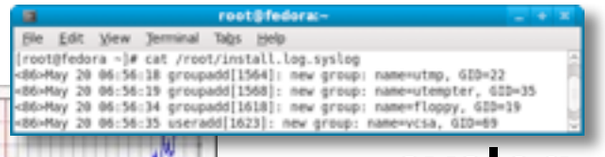
Packet Capture

Traffic Inspection

Attack Detection

NetFlow





syslog

Log Recording

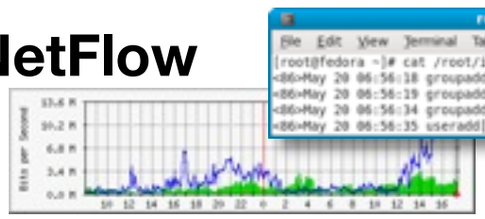
What is Bro?

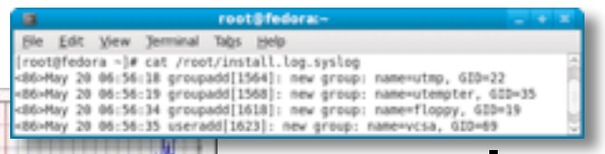
Packet Capture

Traffic Inspection

Attack Detection

NetFlow





syslog

Log Recording

Flexibility
Abstraction
Data Structures

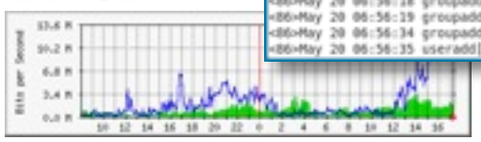
What is Bro?

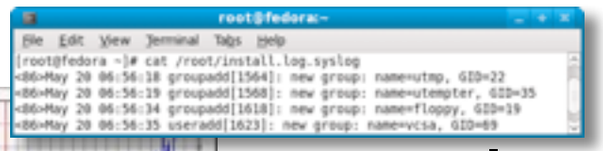
Packet Capture

Traffic Inspection

Attack Detection

NetFlow





syslog

Log Recording

Flexibility
Abstraction
Data Structures

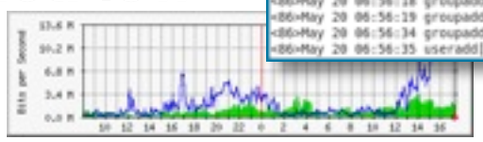
What is Bro?

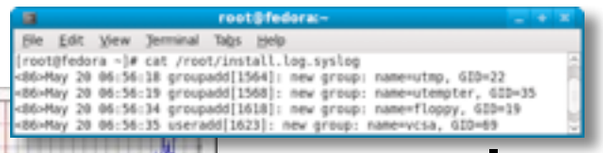
Packet Capture

Traffic Inspection

Attack Detection

NetFlow





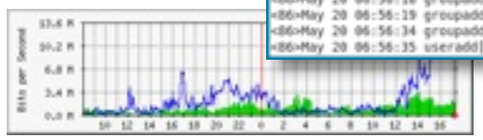
syslog

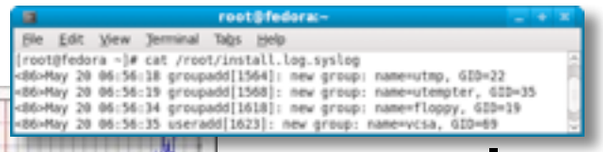
Log Recording

Flexibility
Abstraction
Data Structures

What is Bro?

NetFlow





syslog

Packet Capture

Traffic Inspection

Attack Detection

Log Recording

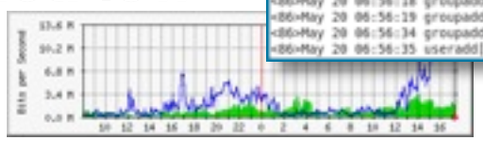
Flexibility
Abstraction
Data Structures

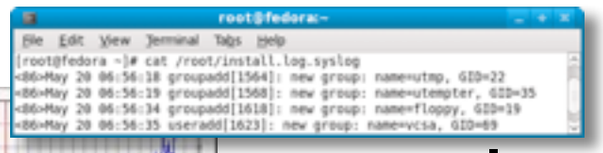
“Domain-specific Python”

NCSA

What is Bro?

NetFlow





syslog

Packet Capture

Traffic Inspection

Attack Detection

Log Recording

Flexibility
Abstraction
Data Structures

Sum is more than the pieces

"Domain-specific Python"

NCSA

Philosophy

Philosophy

Fundamentally different from other IDS.

Reset your idea of an IDS before starting to use Bro.

Philosophy

Fundamentally different from other IDS.

Reset your idea of an IDS before starting to use Bro.

Real-time network analysis *framework.*

Primarily an IDS, but many use it for general traffic analysis.

Philosophy

Fundamentally different from other IDS.

Reset your idea of an IDS before starting to use Bro.

Real-time network analysis *framework.*

Primarily an IDS, but many use it for general traffic analysis.

Policy-neutral at the core.

Can accommodate a range of detection approaches.

Philosophy

Fundamentally different from other IDS.

Reset your idea of an IDS before starting to use Bro.

Real-time network analysis *framework.*

Primarily an IDS, but many use it for general traffic analysis.

Policy-neutral at the core.

Can accommodate a range of detection approaches.

Highly stateful.

Tracks extensive application-layer network state.

Philosophy

Fundamentally different from other IDS.

Reset your idea of an IDS before starting to use Bro.

Real-time network analysis *framework.*

Primarily an IDS, but many use it for general traffic analysis.

Policy-neutral at the core.

Can accommodate a range of detection approaches.

Highly stateful.

Tracks extensive application-layer network state.

Supports forensics.

Extensively logs what it sees.

Target Audience

Target Audience

Network-savvy users.

Requires understanding of your network.

Target Audience

Network-savvy users.

Requires understanding of your network.

Unixy mindset.

Command-line based, fully customizable.

Target Audience

Network-savvy users.

Requires understanding of your network.

Unixy mindset.

Command-line based, fully customizable.

Large-scale environments.

Effective also with liberal security policies.

Bro History

1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013

Vern writes 1st line of code

Bro History

1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013

Vern writes 1st line of code

LBNL starts using Bro operationally

v0.2
1st CHANGES entry

v0.4
HTTP analysis
Scan detector
IP fragments
Linux support

v0.6
RegExps
Login analysis

v0.7a48
Consistent CHANGES

v0.7a90
Profiling
State Mgmt

v0.7a175/0.8aX
Signatures
SMTP
IPv6 support
User manual

0.8a37
Communication
Persistence
Namespaces
Log Rotation

v0.8aX/0.9aX
SSL/SMB
STABLE releases
BroLite

v1.0
BinPAC
IRC/RPC analyzers
64-bit support
Sane version numbers

v1.1/v1.2
when Stmt
Resource tuning
Broccoli
DPD

v1.3
Ctor expressions
GeoIP
Conn Compressor

v1.4
DHCP/BitTorrent
HTTP entities
NetFlow
Bro Lite Deprecated

v1.5
BroControl

Bro SDCI

v2.0
New Scripts

v2.1
IPv6
Input Framew.

v2.2 (beta)
File Analysis
Summary Stat.

Bro History

Academic Publications

1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013

- 1998: USENIX Paper
- 2000: Stepping Stone Detector
- 2003: Anonymizer, Active Mapping, Context Signat.
- 2004: TRW, State Mgmt., Independ. State
- 2005: Host Context, Time Machine, Enterprise Traffic
- 2006: BinPAC, DPD, 2nd Path
- 2007: Bro Cluster, Shunt
- 2008: Autotuning
- 2009: Parallel Prototype
- 2012: Input Framework

- Vern writes 1st line of code
- LBNL starts using Bro operationally
- v0.2 1st CHANGES entry
- v0.4 HTTP analysis, Scan detector, IP fragments, Linux support
- v0.6 RegExps, Login analysis
- v0.7a48 Consistent CHANGES
- v0.7a90 Profiling, State Mgmt
- v0.7a175/0.8aX Signatures, SMTP, IPv6 support, User manual
- 0.8a37 Communication, Persistence, Namespaces, Log Rotation
- v0.8aX/0.9aX SSL/SMB, STABLE releases, BroLite
- v1.0 BinPAC, IRC/RPC analyzers, 64-bit support, Sane version numbers
- v1.1/v1.2 *when* Stmt, Resource tuning, Broccoli, DPD
- v1.3 Ctor expressions, GeoIP, Conn Compressor
- v1.4 DHCP/BitTorrent, HTTP entities, NetFlow, Bro Lite Deprecated
- v1.5 BroControl
- Bro SDCI
- v2.0 New Scripts
- v2.1 IPv6, Input Framew.
- v2.2 (beta) File Analysis, Summary Stat.

"Who's Using It?"

Installations across the US

Universities
Research Labs
Supercomputer Centers
Fortune 50 Industry

Examples

Lawrence Berkeley National Lab
Indiana University
National Center for Supercomputing Applications
National Center for Atmospheric Research

... and many more sites

Recent User Meetings

Bro Workshop 2011 at NCSA
Bro Exchange 2012 at NCAR
Bro Exchange 2013 at NCSA

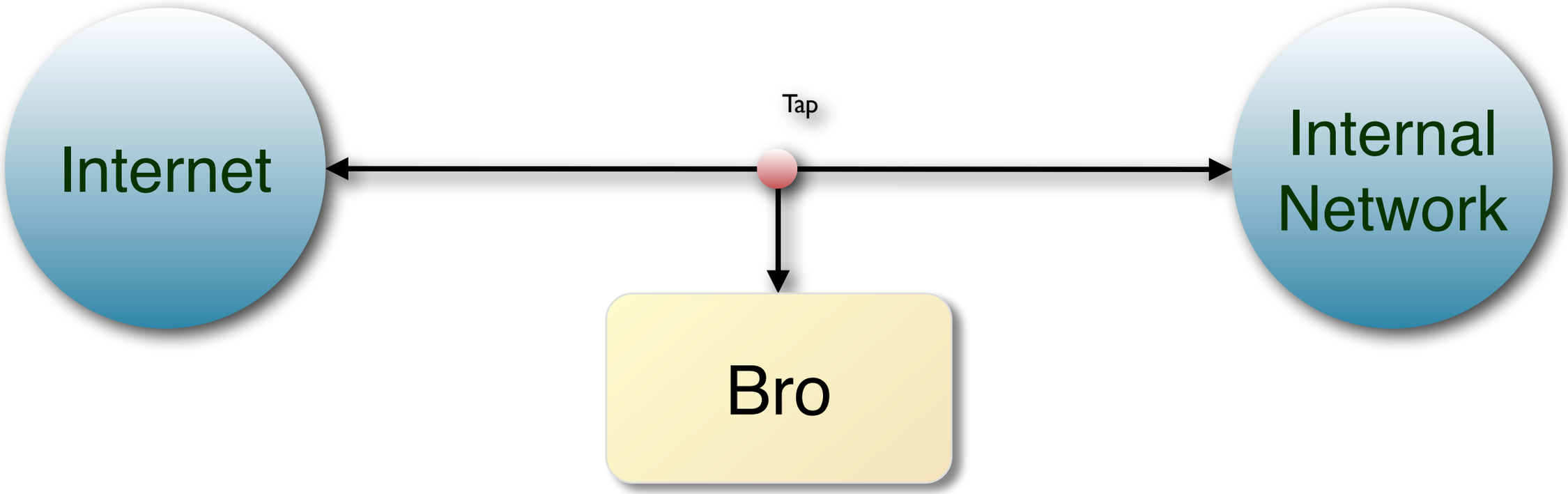
Each attended by about 50-90 operators from
from 30-50 organizations

Fully integrated into *Security Onion*

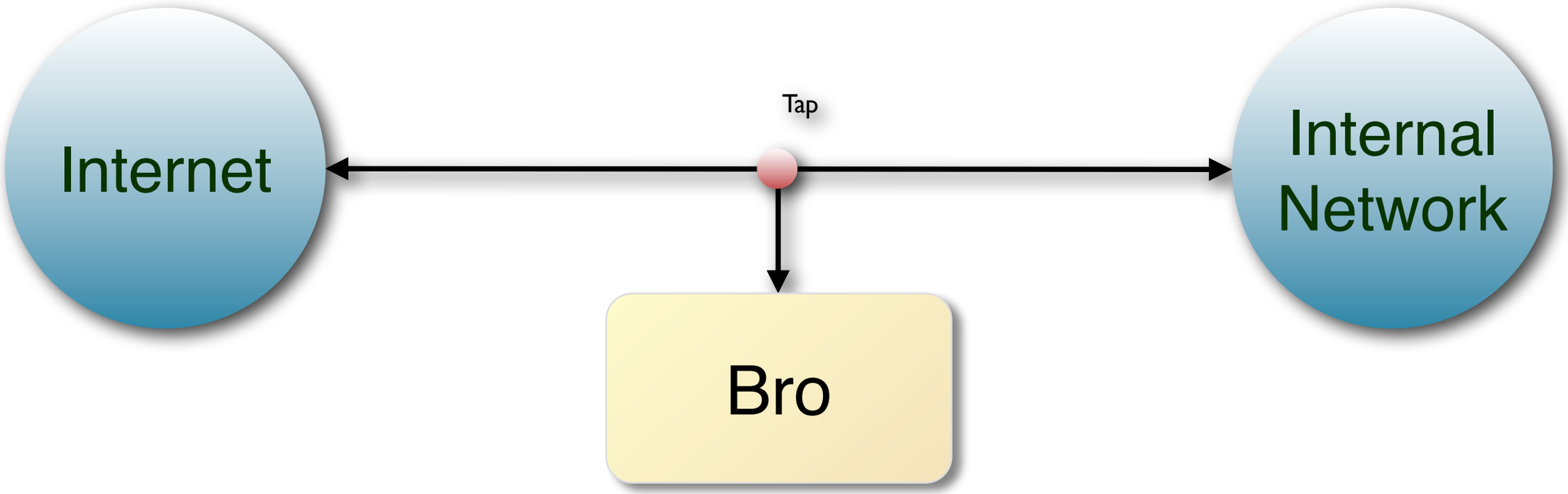
Popular security-oriented Linux distribution

Deployment

Deployment



Deployment



Runs on commodity platforms.

- Standard PCs & NICs.
- Supports FreeBSD/Linux/OS X.

Creating Visibility with Bro

Creating Visibility with Bro

```
> bro -i en0
[ ... wait ...]
> cat conn.log
```

Creating Visibility with Bro

```
> bro -i en0
[ ... wait ...]
> cat conn.log
```

#fields ts	*id.orig_h*	*id.orig_p*	*id.resp_h*	*id.resp_p*	*proto*	*service*	*duration*
1144876741.1198	*192.150.186.169*	*53115*	*82.94.237.218*	*80*	*tcp*	*http*	*16.14929*
1144876612.6063	192.150.186.169	53090	198.189.255.82	80	tcp	http	4.437460
1144876596.5597	192.150.186.169	53051	193.203.227.129	80	tcp	http	0.372440
1144876606.7789	192.150.186.169	53082	198.189.255.73	80	tcp	http	0.597711
1144876741.4693	**192.150.186.169**	**53116**	**82.94.237.218**	**80**	**tcp**	**http**	**16.02667**
1144876745.6102	**192.150.186.169**	**53117**	**66.102.7.99**	**80**	**tcp**	**http**	**1.004346**
1144876605.6847	192.150.186.169	53075	207.151.118.143	80	tcp	http	0.029663

Creating Visibility with Bro

```
> bro -i en0
[ ... wait ...]
> cat conn.log
```

#fields ts	*id.orig_h*	*id.orig_p*	*id.resp_h*	*id.resp_p*	*proto*	*service*	*duration*
1144876741.1198	*192.150.186.169*	*53115*	*82.94.237.218*	*80*	*tcp*	*http*	*16.14929*
1144876612.6063	192.150.186.169	53090	198.189.255.82	80	tcp	http	4.437460
1144876596.5597	192.150.186.169	53051	193.203.227.129	80	tcp	http	0.372440
1144876606.7789	192.150.186.169	53082	198.189.255.73	80	tcp	http	0.597711
1144876741.4693	**192.150.186.169**	**53116**	**82.94.237.218**	**80**	**tcp**	**http**	**16.02667**
1144876745.6102	**192.150.186.169**	**53117**	**66.102.7.99**	**80**	**tcp**	**http**	**1.004346**
1144876605.6847	192.150.186.169	53075	207.151.118.143	80	tcp	http	0.029663

```
> cat http.log
```

Creating Visibility with Bro

```
> bro -i en0
[ ... wait ...]
> cat conn.log
```

#fields ts	*id.orig_h*	*id.orig_p*	*id.resp_h*	*id.resp_p*	*proto*	*service*	*duration*
1144876741.1198	*192.150.186.169*	*53115*	*82.94.237.218*	*80*	*tcp*	*http*	*16.14929*
1144876612.6063	192.150.186.169	53090	198.189.255.82	80	tcp	http	4.437460
1144876596.5597	192.150.186.169	53051	193.203.227.129	80	tcp	http	0.372440
1144876606.7789	192.150.186.169	53082	198.189.255.73	80	tcp	http	0.597711
1144876741.4693	**192.150.186.169**	**53116**	**82.94.237.218**	**80**	**tcp**	**http**	**16.02667**
1144876745.6102	**192.150.186.169**	**53117**	**66.102.7.99**	**80**	**tcp**	**http**	**1.004346**
1144876605.6847	192.150.186.169	53075	207.151.118.143	80	tcp	http	0.029663

```
> cat http.log
```

#fields ts	*id.orig_h*	*id.orig_p [...]*	*host*	*uri*	*status_code*	*user_agent [...]*
1144876741.6335	192.150.186.169	53116	docs.python.org	/lib/lib.css	200	Mozilla/5.0
1144876742.1687	192.150.186.169	53116	docs.python.org	/icons/previous.png	304	Mozilla/5.0
1144876741.2838	192.150.186.169	53115	docs.python.org	/lib/lib.html	200	Mozilla/5.0
1144876742.3337	192.150.186.169	53116	docs.python.org	/icons/up.png	304	Mozilla/5.0
1144876742.3337	192.150.186.169	53116	docs.python.org	/icons/next.png	304	Mozilla/5.0
1144876742.3337	192.150.186.169	53116	docs.python.org	/icons/contents.png	304	Mozilla/5.0
1144876742.3337	192.150.186.169	53116	docs.python.org	/icons/modules.png	304	Mozilla/5.0
1144876742.3338	192.150.186.169	53116	docs.python.org	/icons/index.png	304	Mozilla/5.0
1144876745.6144	192.150.186.169	53117	www.google.com	/	200	Mozilla/5.0

Creating Visibility with Bro

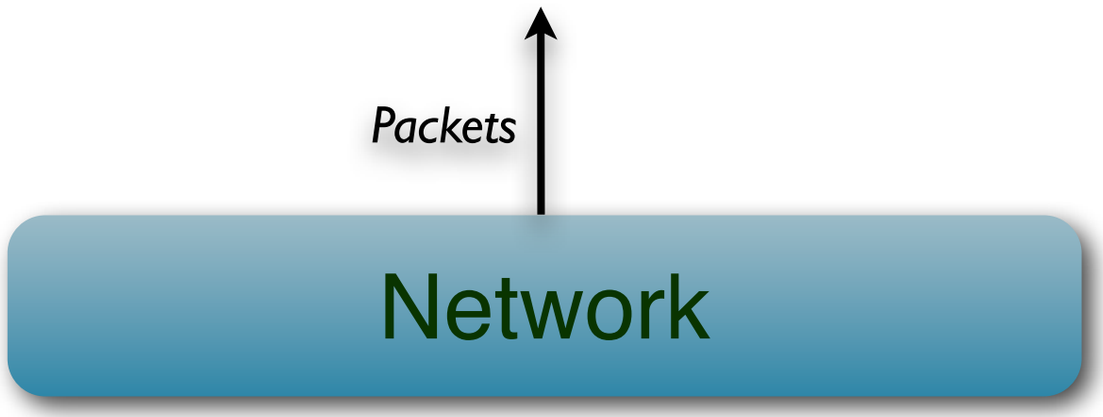
```
> bro -i en0
[ ... wait ...]
> cat conn.log
```

```
#fields ts          id.orig_h         id.orig_p   id.resp_h         id.resp_p proto   service   duration
1144876741.1198  192.150.186.169 53115       82.94.237.218   80        tcp     http      16.14929
1144876612.6063  192.150.186.169 53090       198.189.255.82  80        tcp     http      4.437460
```

```
[...] host                uri                       status_code  user_agent [...]
      docs.python.org     /lib/lib.css              200          Mozilla/5.0
      docs.python.org     /icons/previous.png       304          Mozilla/5.0
      docs.python.org     /lib/lib.html             200          Mozilla/5.0
      docs.python.org     /icons/up.png             304          Mozilla/5.0
      docs.python.org     /icons/next.png           304          Mozilla/5.0
      docs.python.org     /icons/contents.png       304          Mozilla/5.0
      docs.python.org     /icons/modules.png        304          Mozilla/5.0
      docs.python.org     /icons/index.png          304          Mozilla/5.0
       www.google.com     /                         200          Mozilla/5.0
```

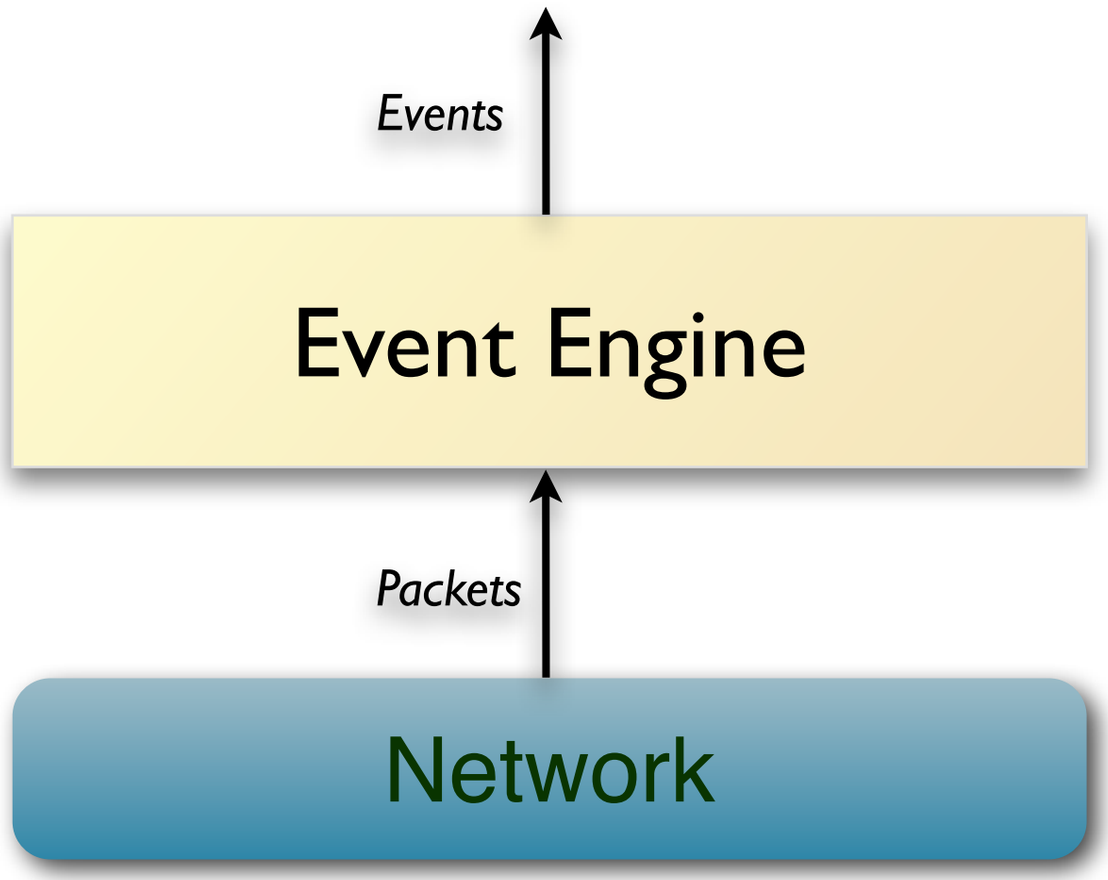
```
1144876742.3338  192.150.186.169  53116        docs.python.org  /icons/index.png     304          Mozilla/5.0
1144876745.6144  192.150.186.169  53117         www.google.com  /                    200          Mozilla/5.0
```

Architecture



Architecture

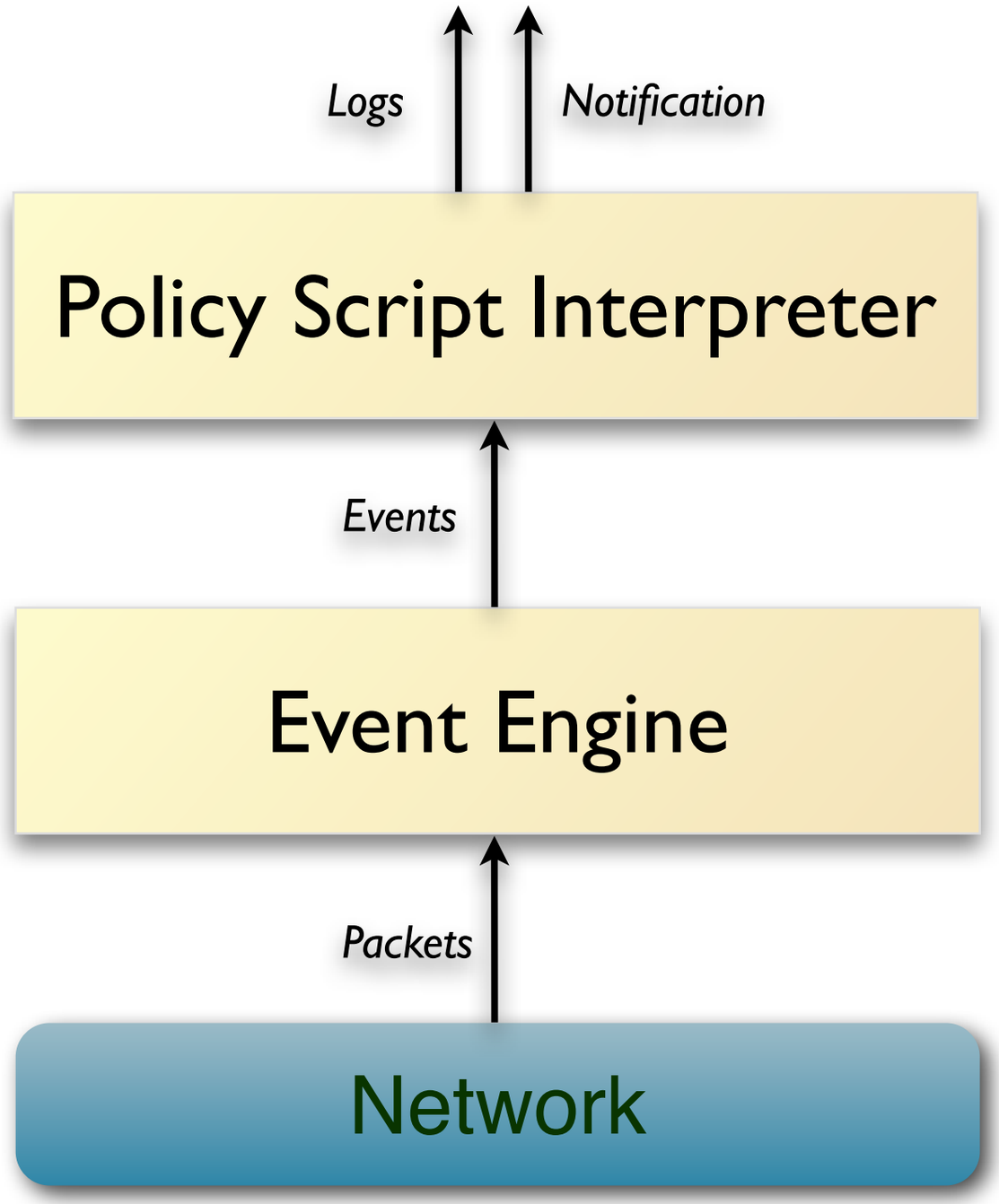
Protocol Decoding



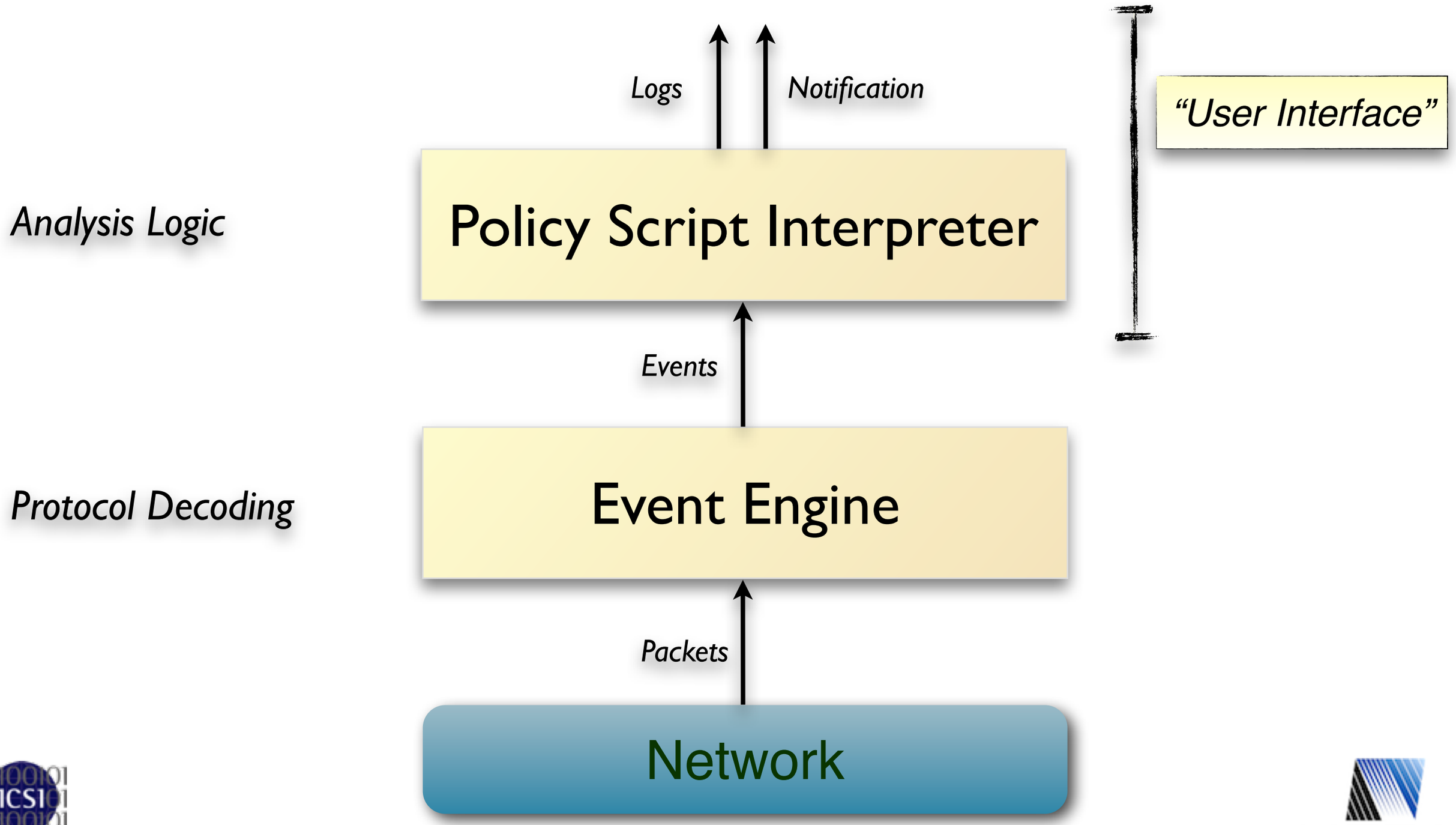
Architecture

Analysis Logic

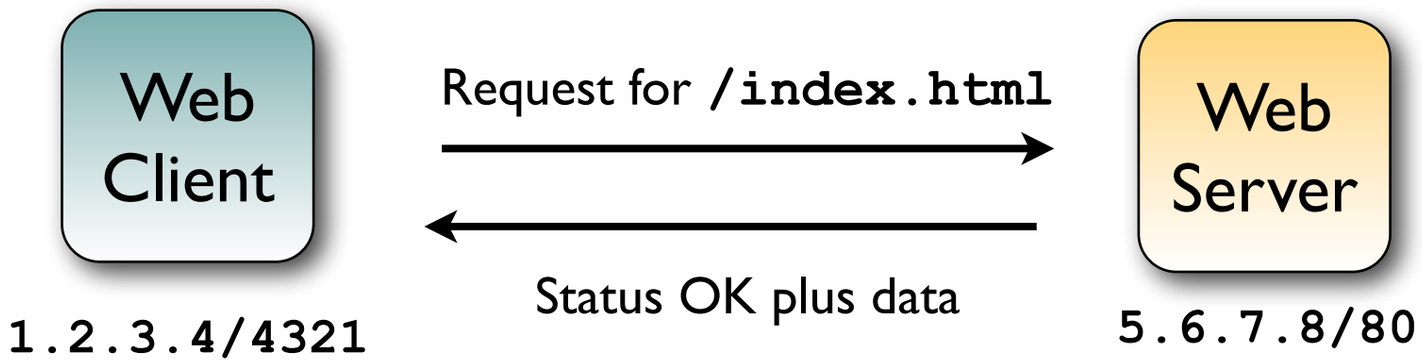
Protocol Decoding



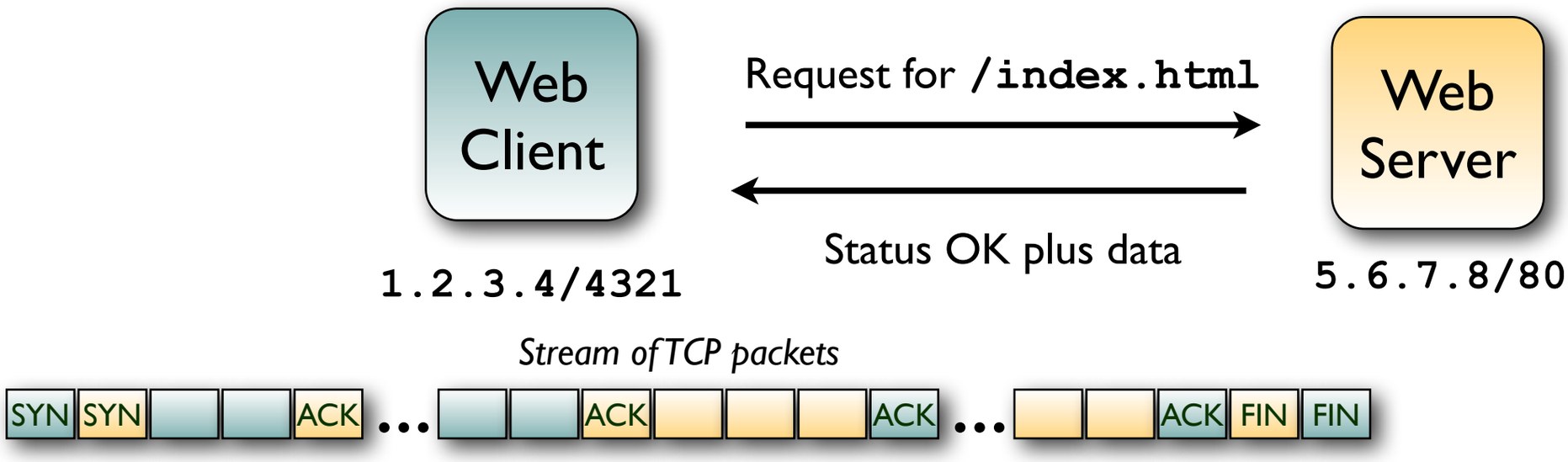
Architecture



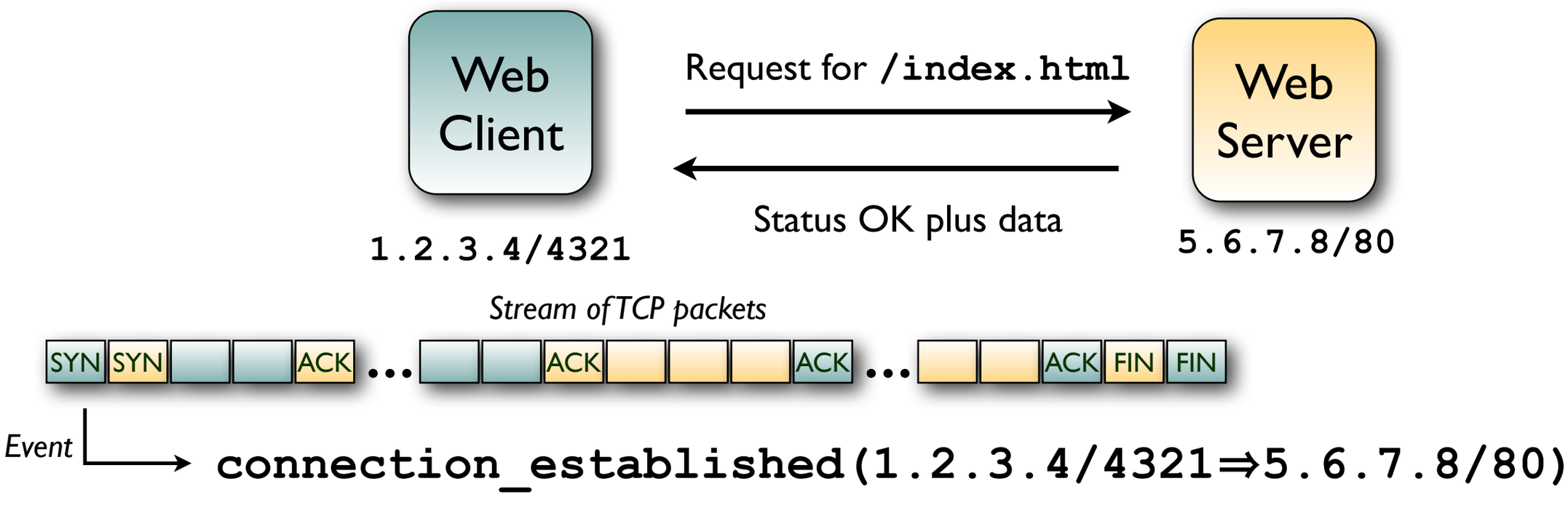
Event Model


Web Client
Request for /index.html
Web Server
Status OK plus data
1.2.3.4/4321
5.6.7.8/80

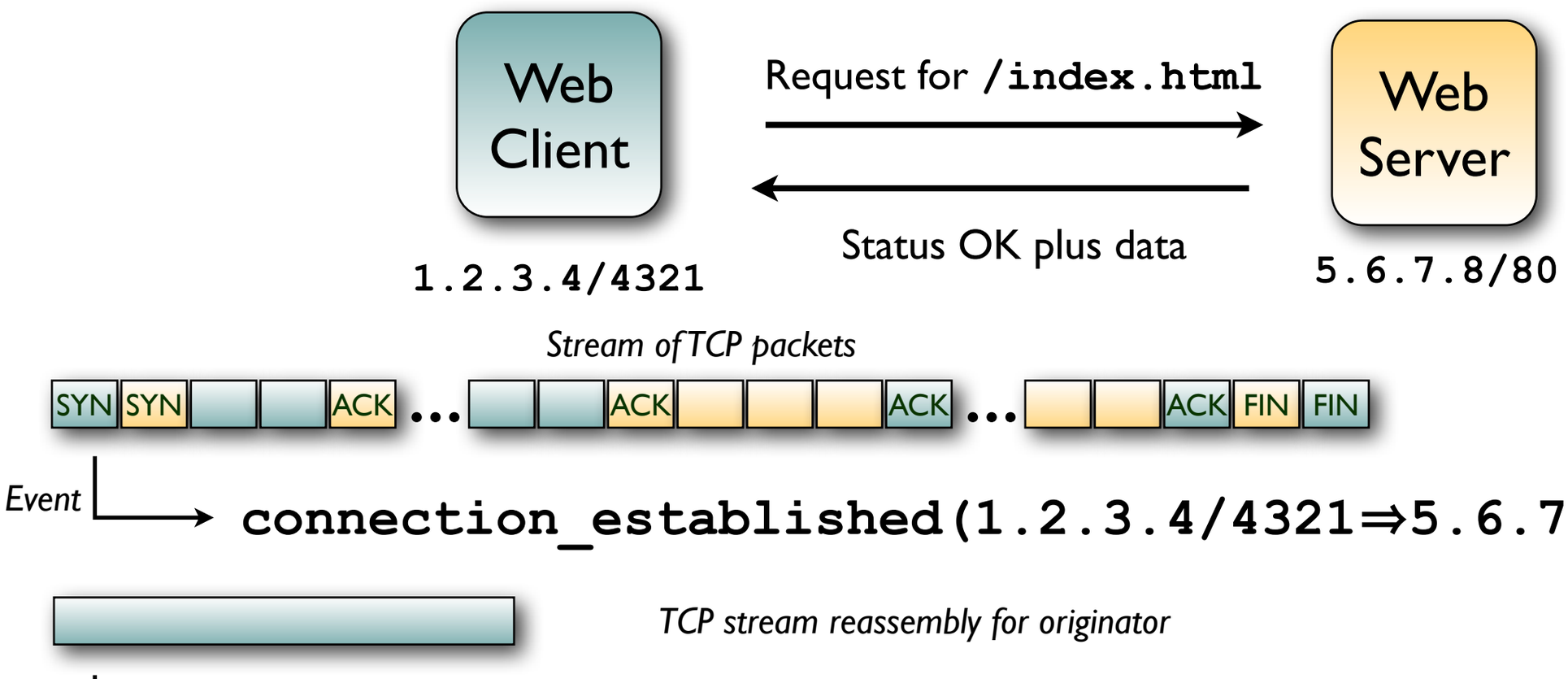
Event Model


Web Client
Request for /index.html
Web Server
Status OK plus data
1.2.3.4/4321
5.6.7.8/80
Stream of TCP packets
SYN
SYN
ACK
ACK
ACK
ACK
FIN
FIN

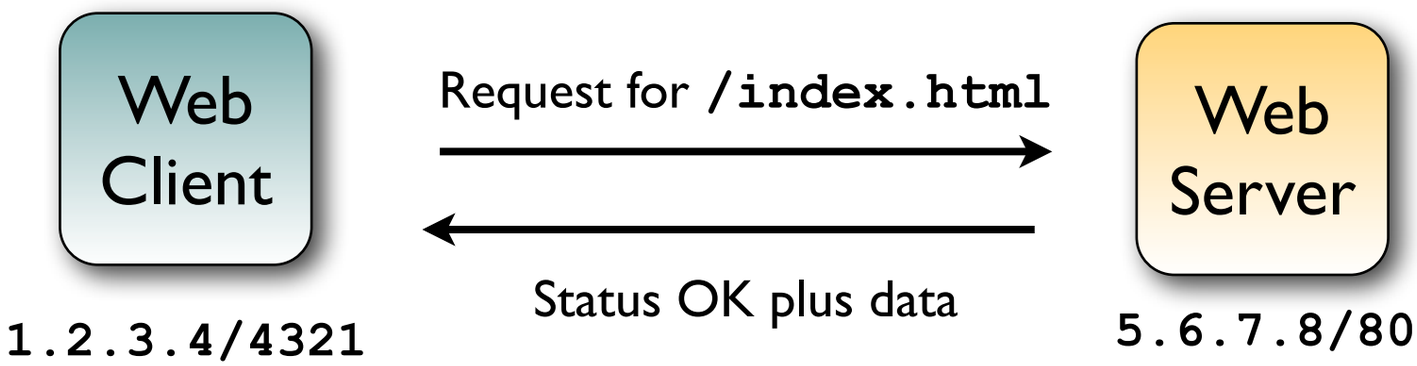
Event Model


Web Client
Request for /index.html
Web Server
Status OK plus data
1.2.3.4/4321
5.6.7.8/80
Stream of TCP packets
SYN
SYN
ACK
ACK
ACK
ACK
FIN
FIN
Event
connection established(1.2.3.4/4321⇒5.6.7.8/80)

Event Model


Web Client
Request for /index.html
Web Server
Status OK plus data
1.2.3.4/4321
5.6.7.8/80
Stream of TCP packets
SYN
SYN
ACK
ACK
ACK
ACK
FIN
FIN
Event
connection established(1.2.3.4/4321⇒5.6.7.8/80)
TCP stream reassembly for originator
Event
http_request(1.2.3.4/4321⇒5.6.7.8/80, "GET", "/index.html")

Event Model



Stream of TCP packets

Event → `connection established(1.2.3.4/4321⇒5.6.7.8/80)`

TCP stream reassembly for originator

Event → `http_request(1.2.3.4/4321⇒5.6.7.8/80, "GET", "/index.html")`

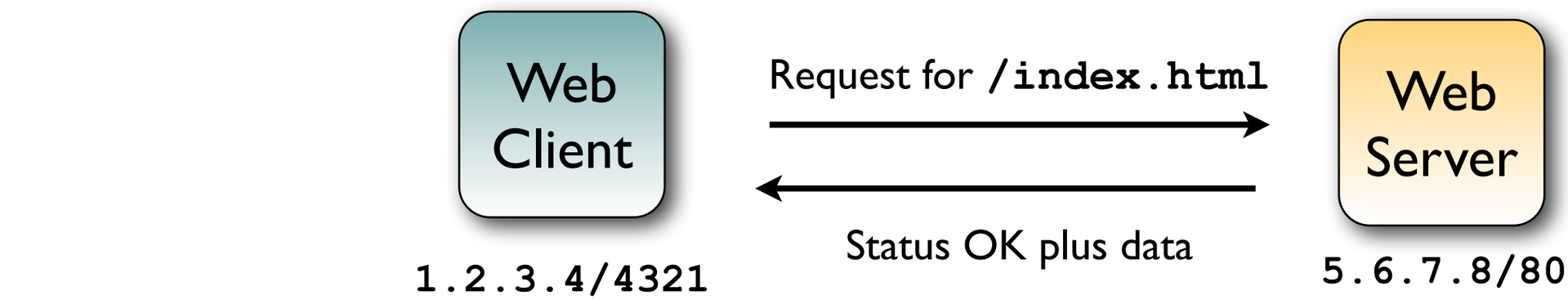
TCP stream reassembly for responder

Event → `http_reply(1.2.3.4/4321⇒5.6.7.8/80, 200, "OK", data)`

ICSI International Computer Science Institute

NCSA

Event Model



Event → `connection_established(1.2.3.4/4321⇒5.6.7.8/80)`

TCP stream reassembly for originator

Event → `http_request(1.2.3.4/4321⇒5.6.7.8/80, "GET", "/index.html")`

TCP stream reassembly for responder

Event → `http_reply(1.2.3.4/4321⇒5.6.7.8/80, 200, "OK", data)`

Event → `connection_finished(1.2.3.4/4321, 5.6.7.8/80)`

ICSI International Computer Science Institute

NCSA

Script Example: Matching URLs

Task: Report all Web requests for files called "`passwd`".

Script Example: Matching URLs

Task: Report all Web requests for files called "`passwd`".

```
event http_request(c: connection,           # Connection.
                   method: string,          # HTTP method.
                   original_URI: string,    # Requested URL.
                   unescaped_URI: string,   # Decoded URL.
                   version: string)         # HTTP version.
{
    if ( method == "GET" && unescaped_URI == /.*passwd/ )
        NOTICE(...); # Alarm.
}
```

Script Example: Scan Detector

Task: Count failed connection attempts per source address.

Script Example: Scan Detector

Task: Count failed connection attempts per source address.

```
global attempts: table[addr] of count &default=0;

event connection_rejected(c: connection)
{
        local source = c$id$orig_h;      # Get source address.
        local n = ++attempts[source];    # Increase counter.
        if ( n == SOME_THRESHOLD )       # Check for threshold.
            NOTICE(...);                 # Alarm.
}
```

Distributed Scripts

Distributed Scripts

Bro comes with >10,000 lines of script code.

Prewritten functionality that's just loaded.

Distributed Scripts

Bro comes with >10,000 lines of script code.

Prewritten functionality that's just loaded.

Scripts generate alarms and logs.

Amendable to extensive customization and extension.

Bro comes with support for ...

Bro comes with support for ...

Extract files from HTTP, SMTP, etc.

Extract/monitor SSL certificates.

Detect malware via Team Cymru's Malware Hash Registry.

Report vulnerable software versions on the network.

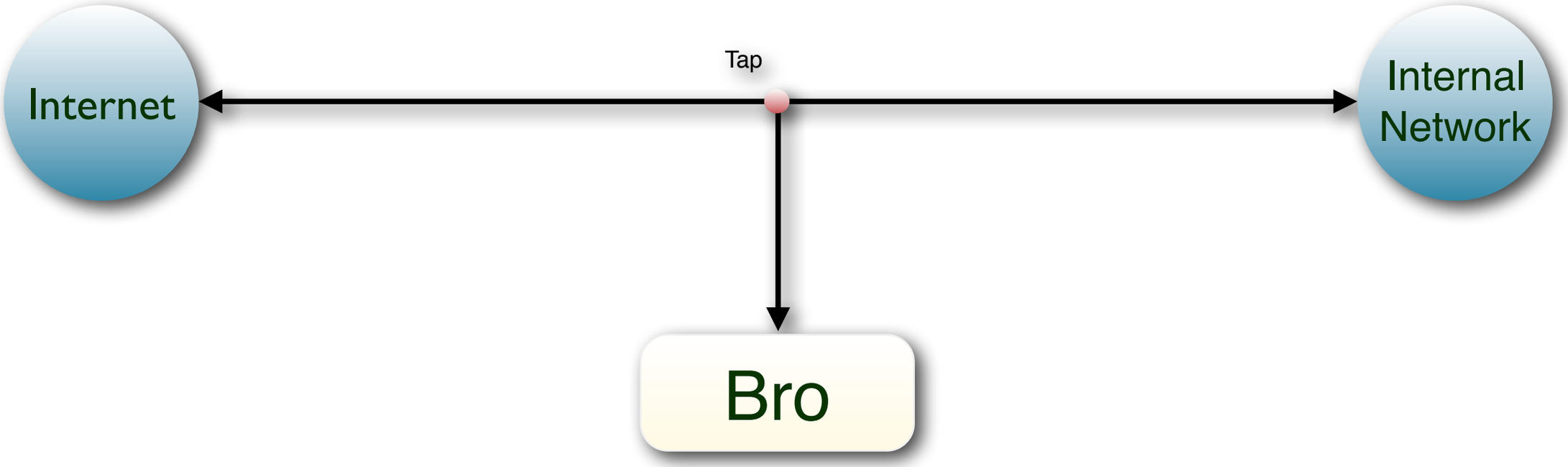
Detect popular web applications.

Detect SSH brute-forcing.

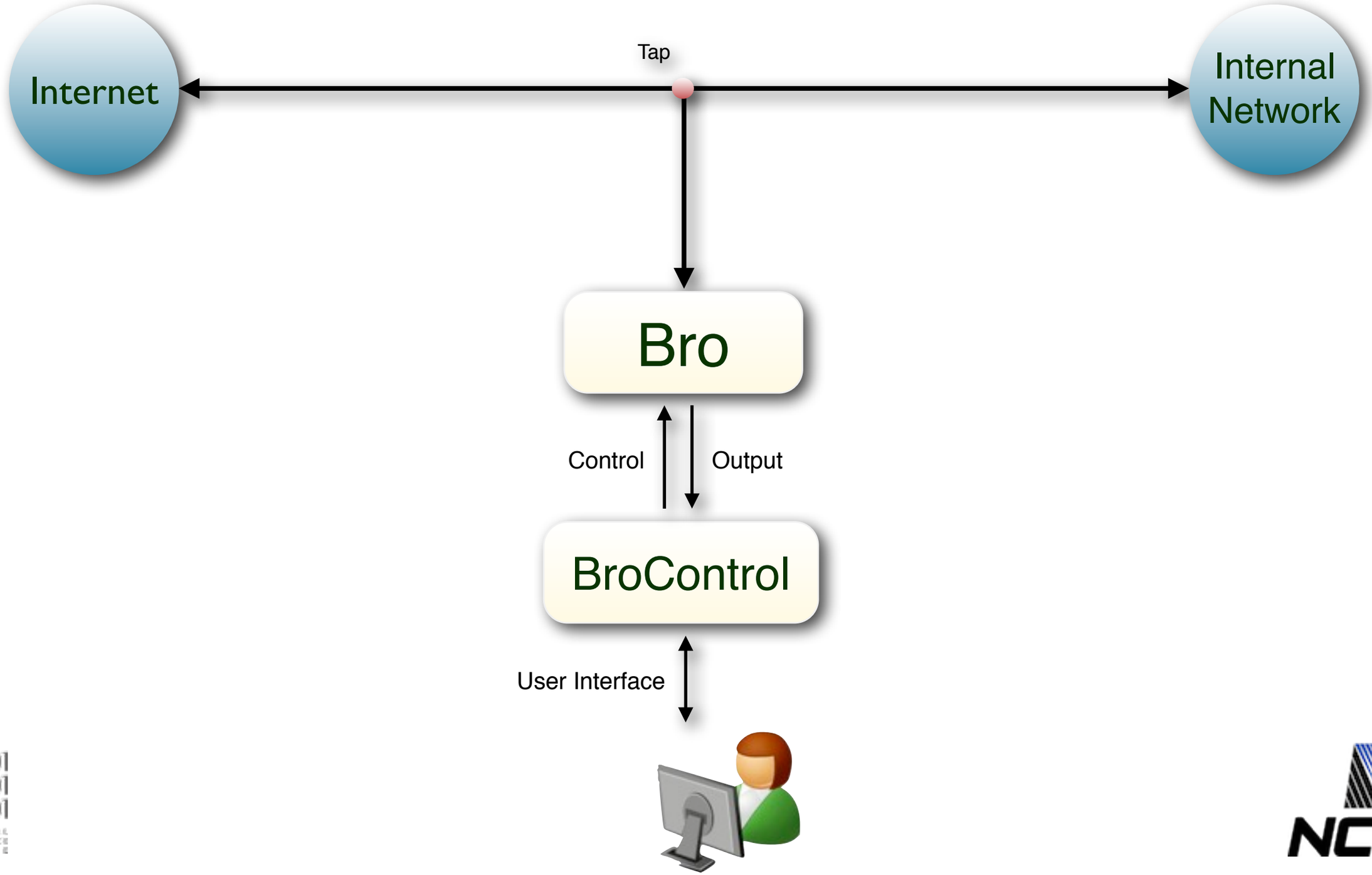
Notable external scripts:

Bro module for Mandiant APT1 report
Lucky 13 detector.
ICSI SSL notary

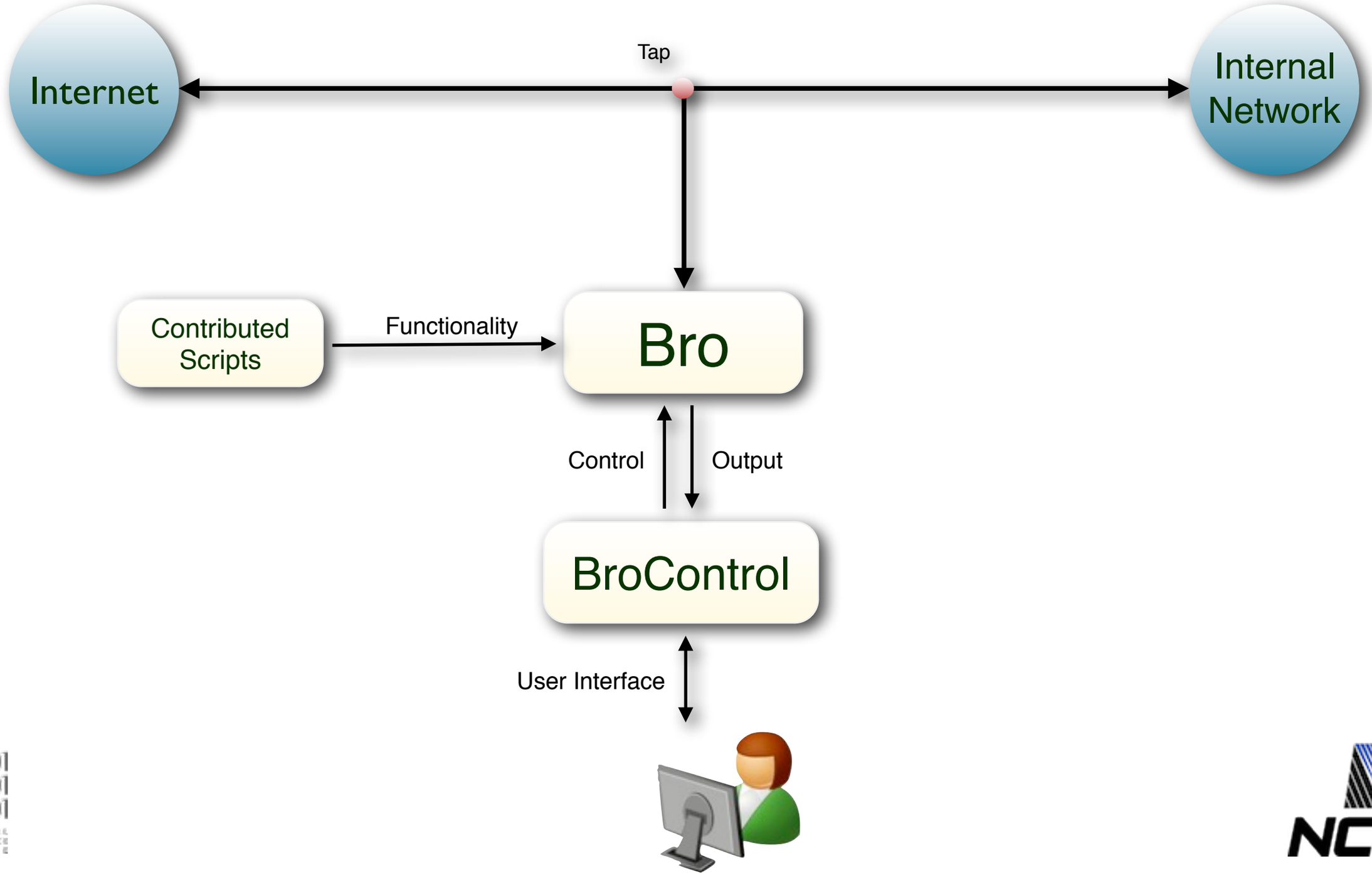
Bro Ecosystem



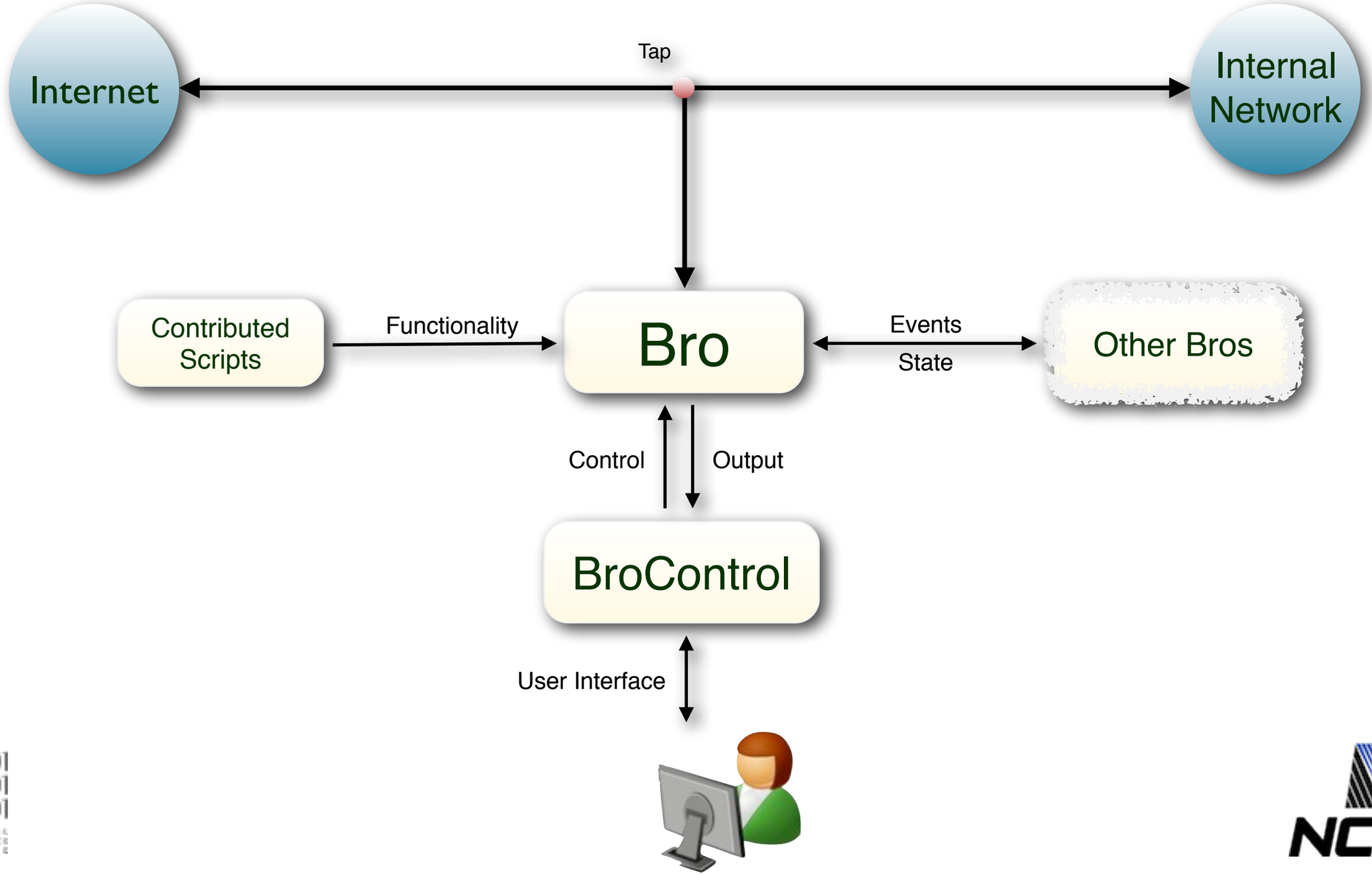
Bro Ecosystem



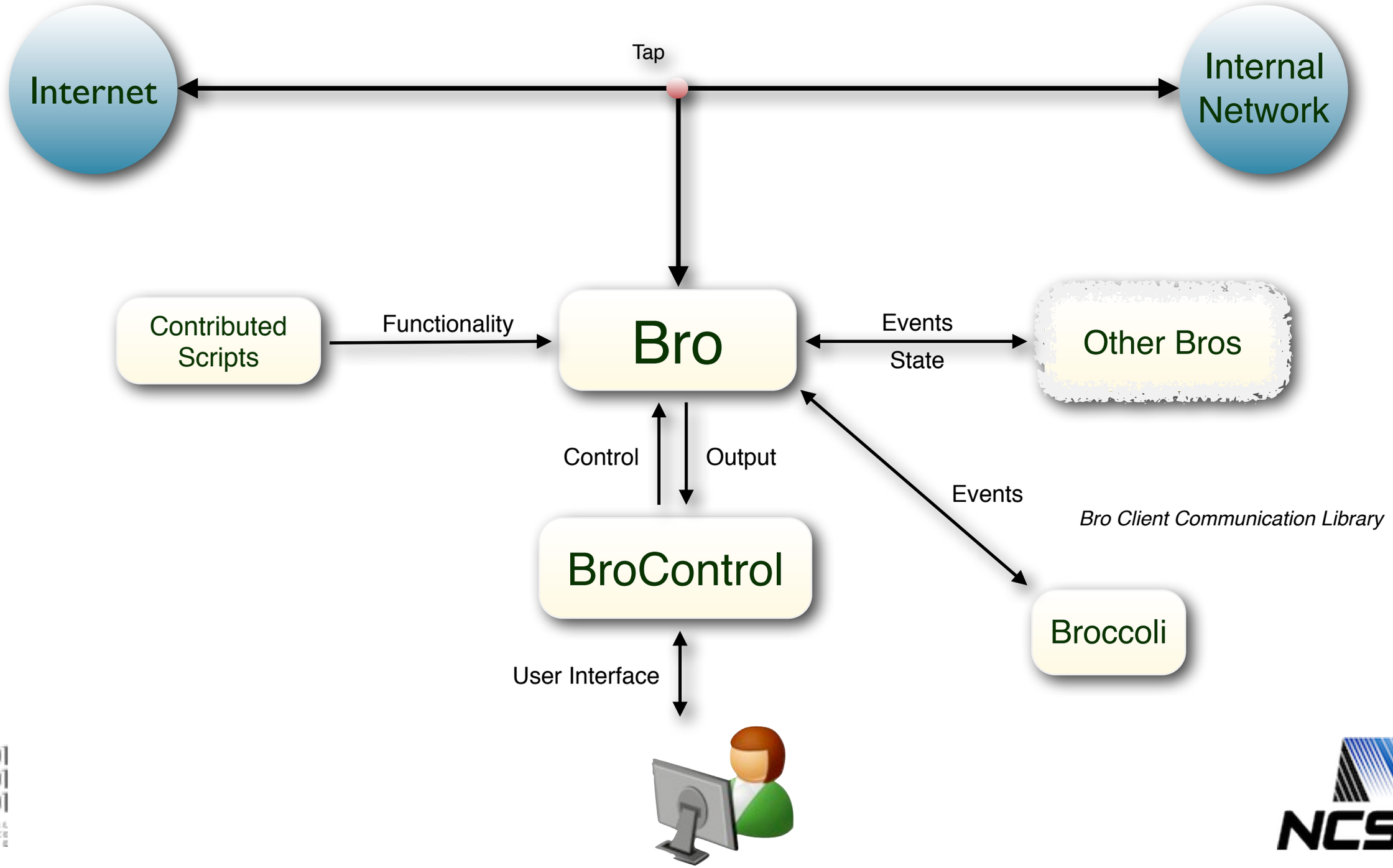
Bro Ecosystem



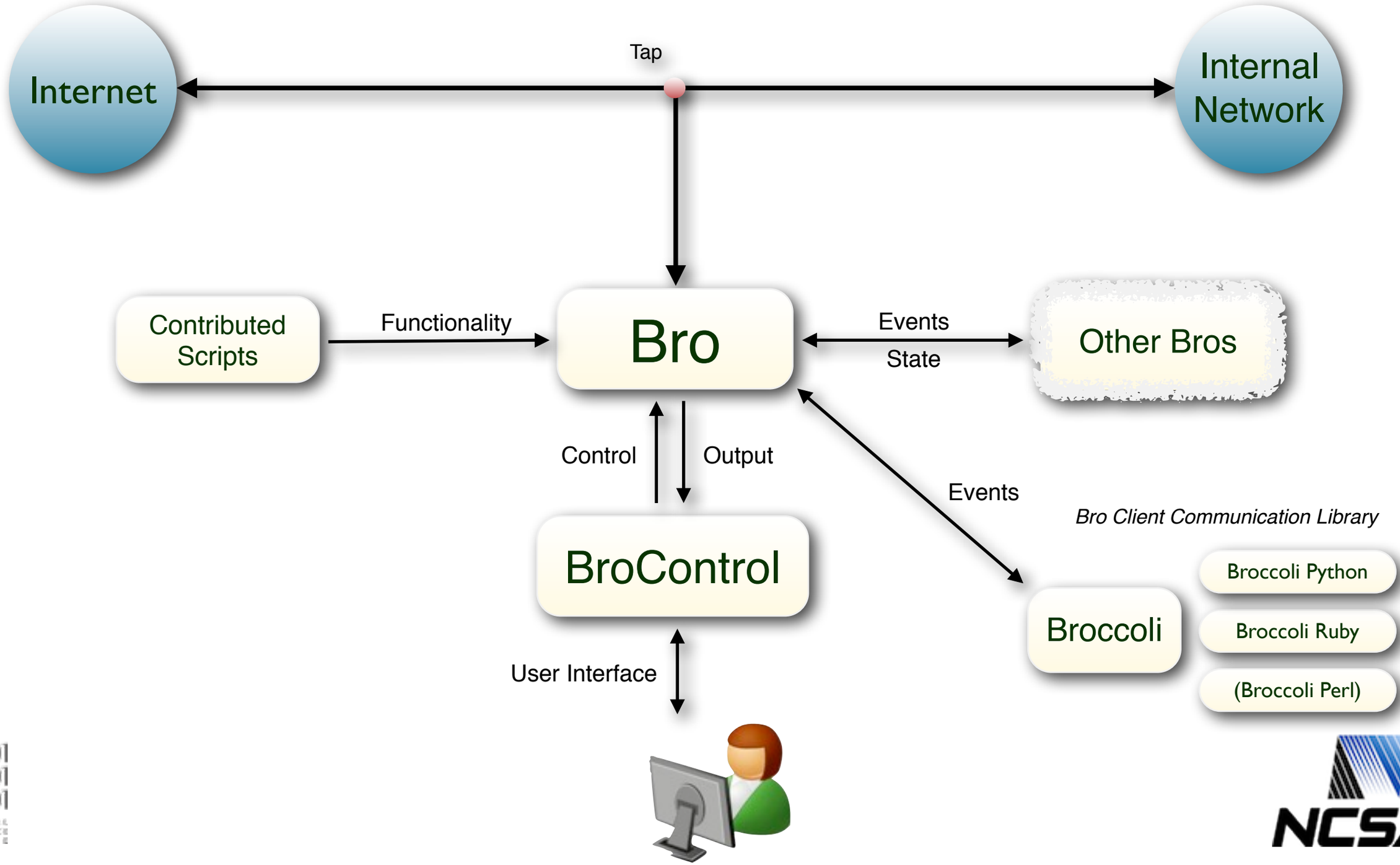
Bro Ecosystem



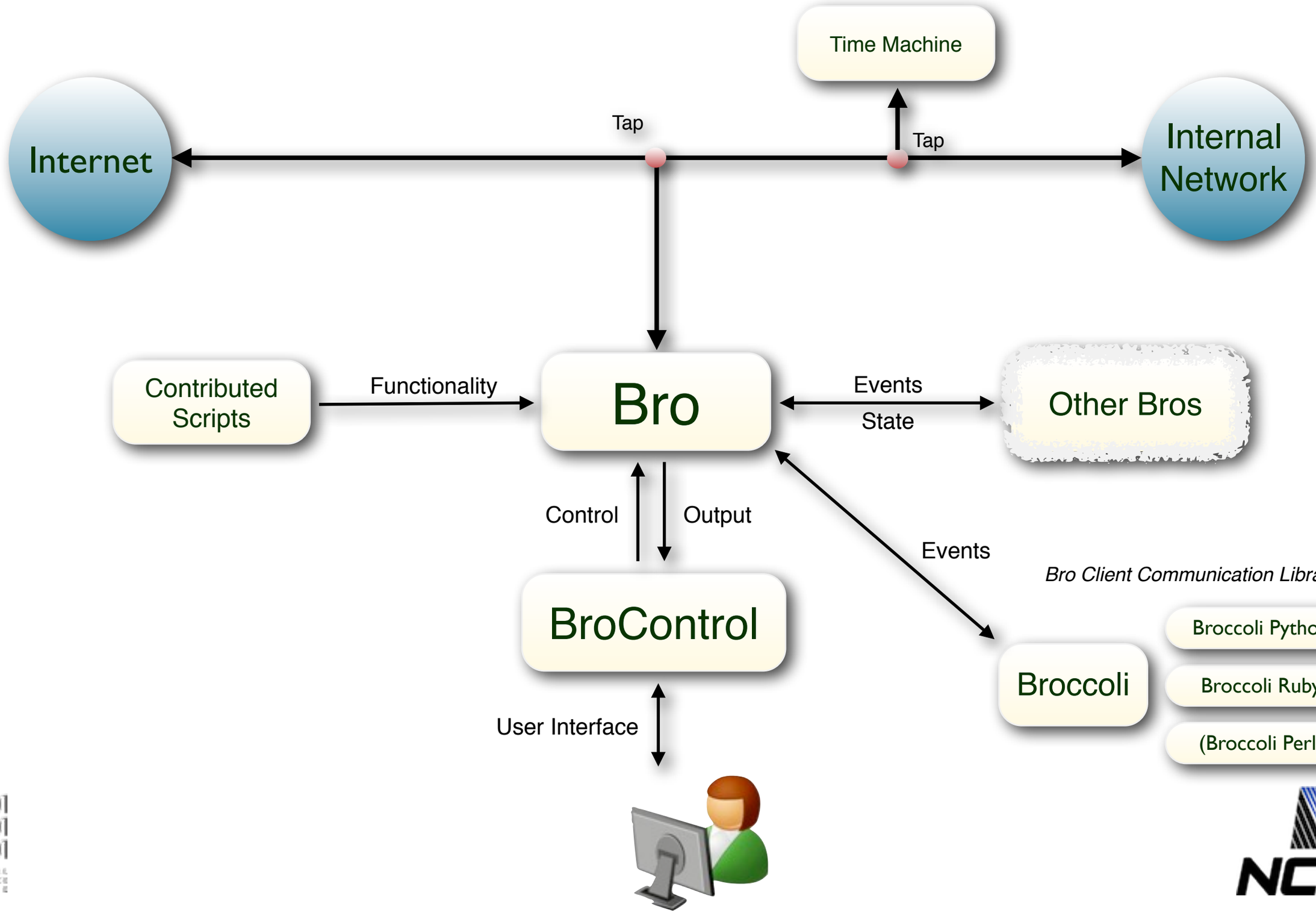
Bro Ecosystem



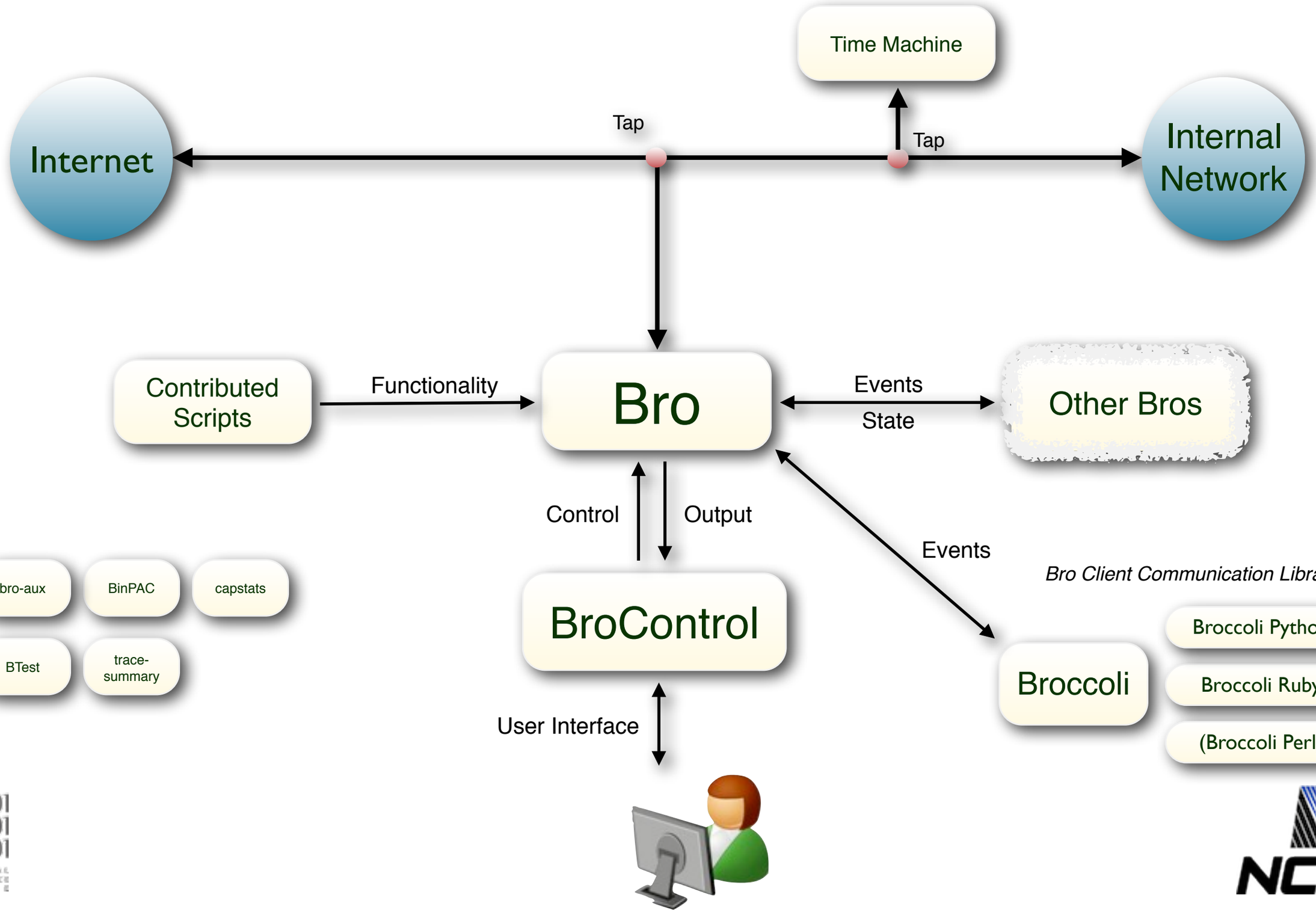
Bro Ecosystem



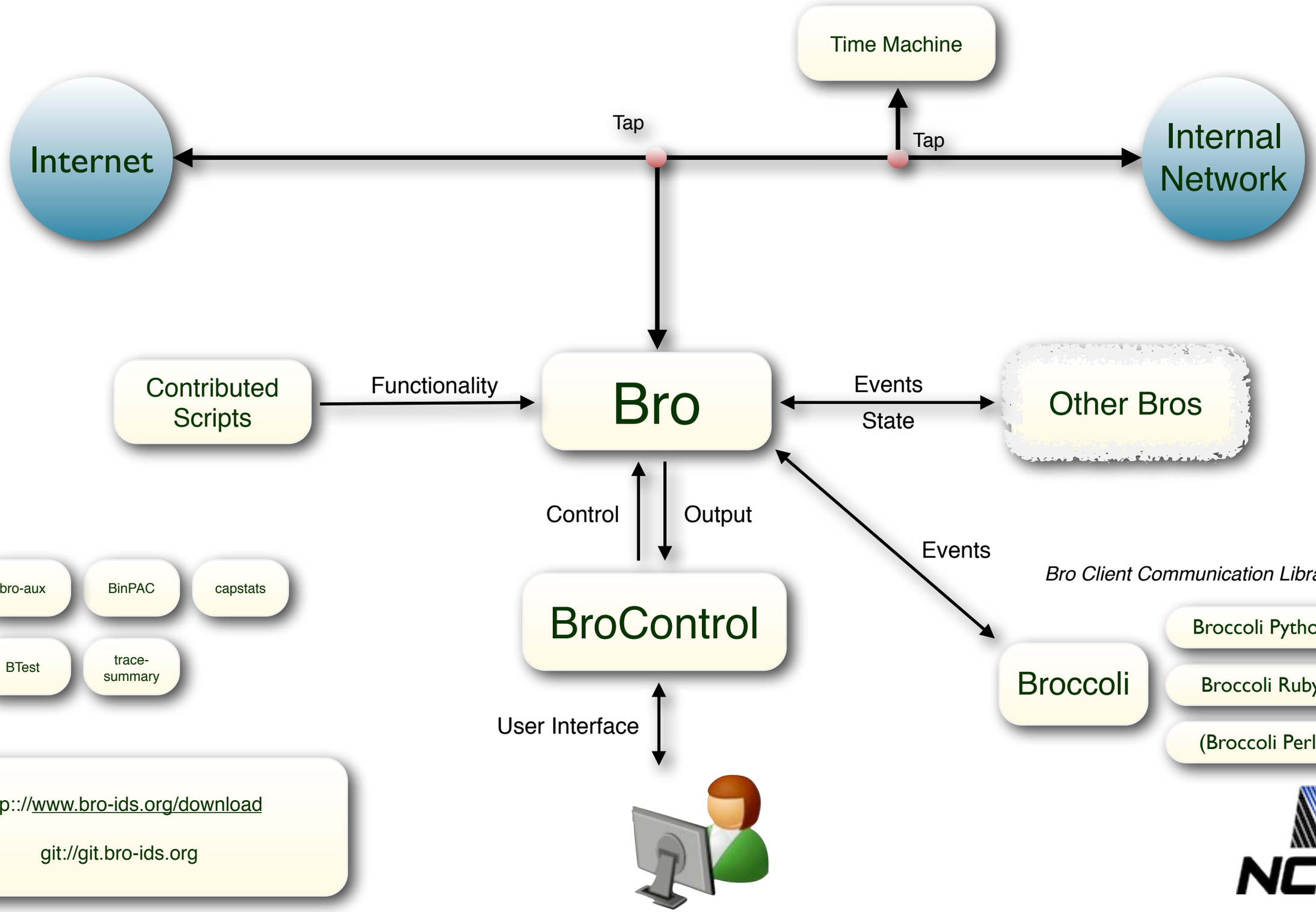
Bro Ecosystem



Bro Ecosystem



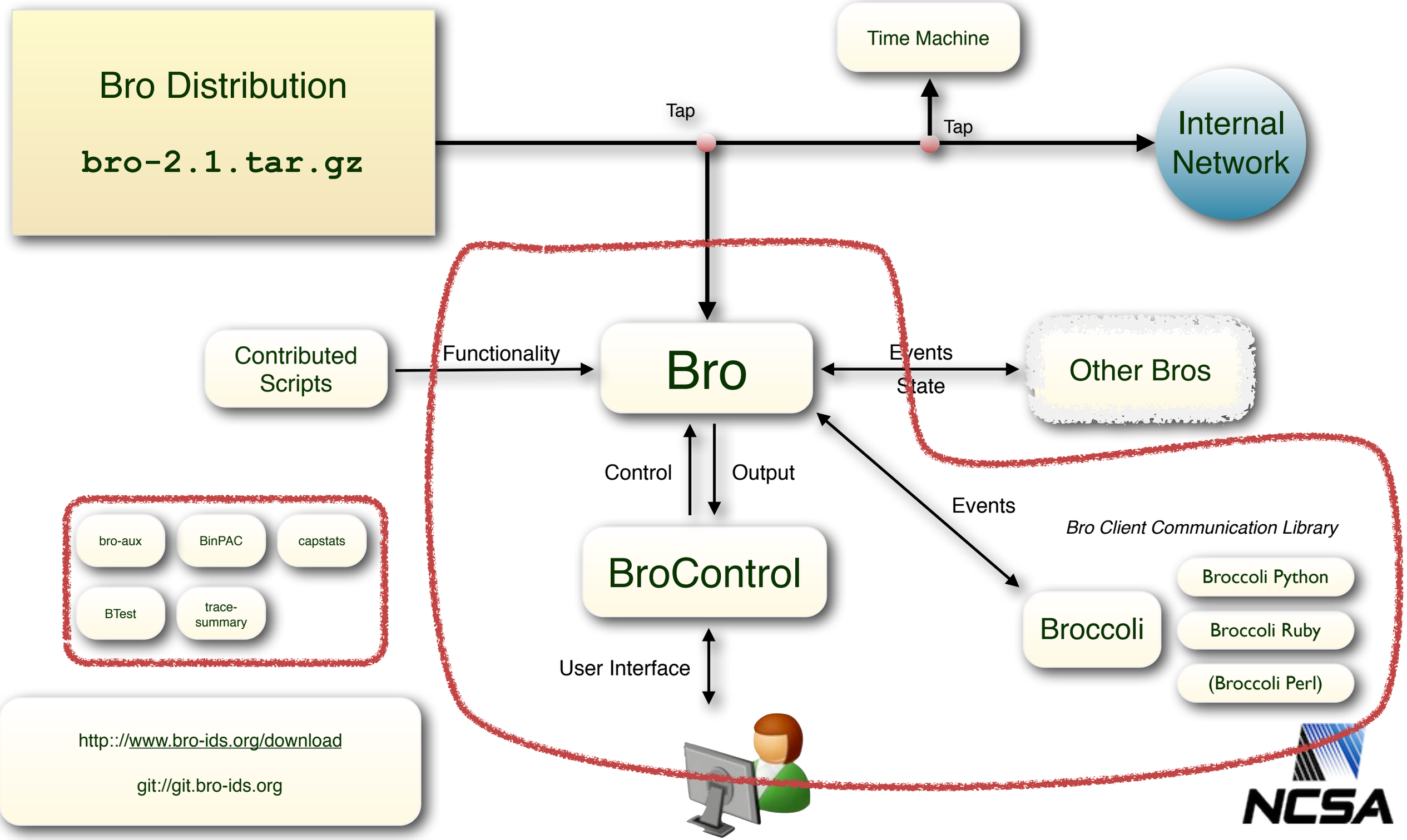
Bro Ecosystem



Bro Ecosystem

Bro Distribution

`bro-2.1.tar.gz`

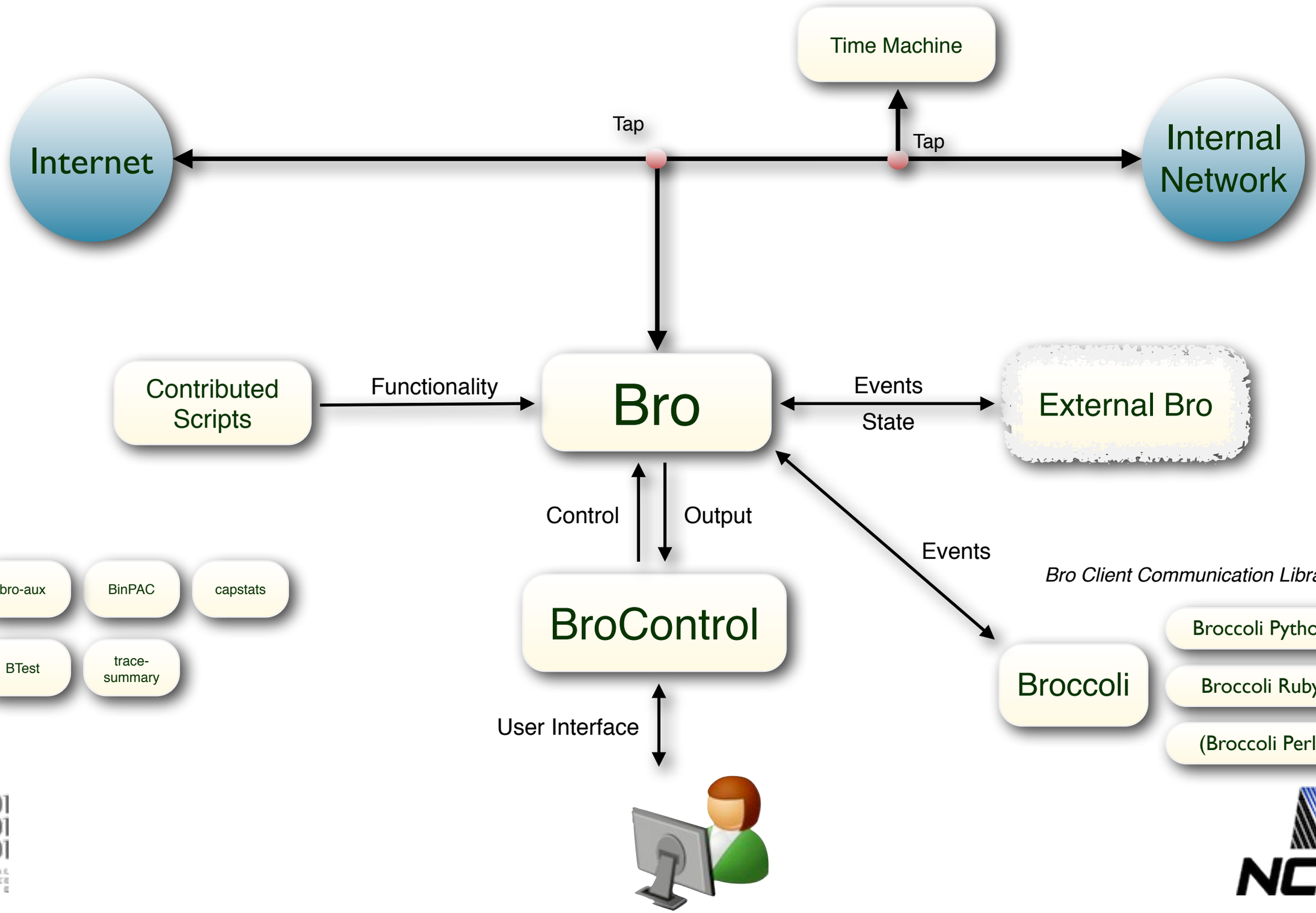


http:://www.bro-ids.org/download

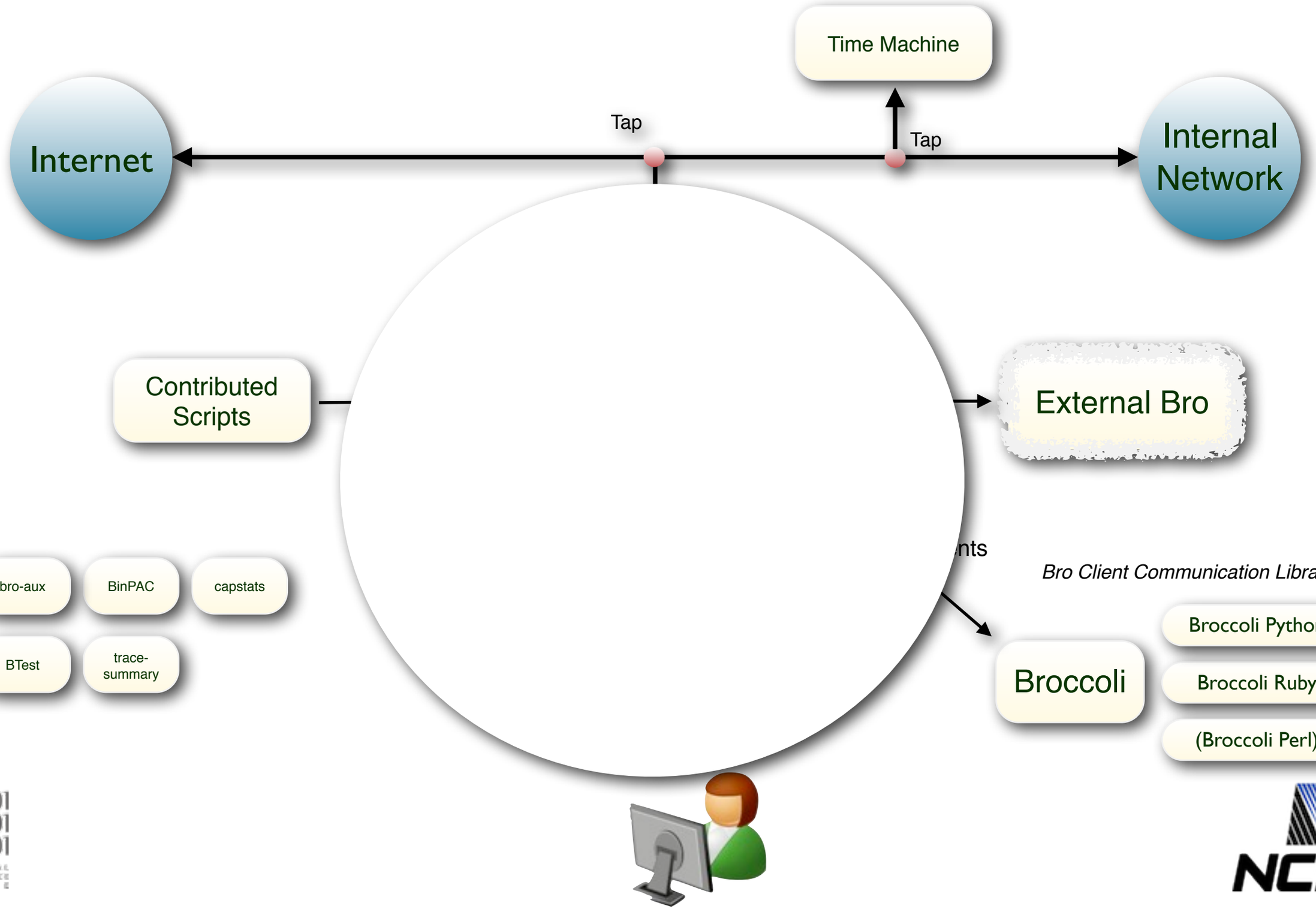
git://git.bro-ids.org

NCSA

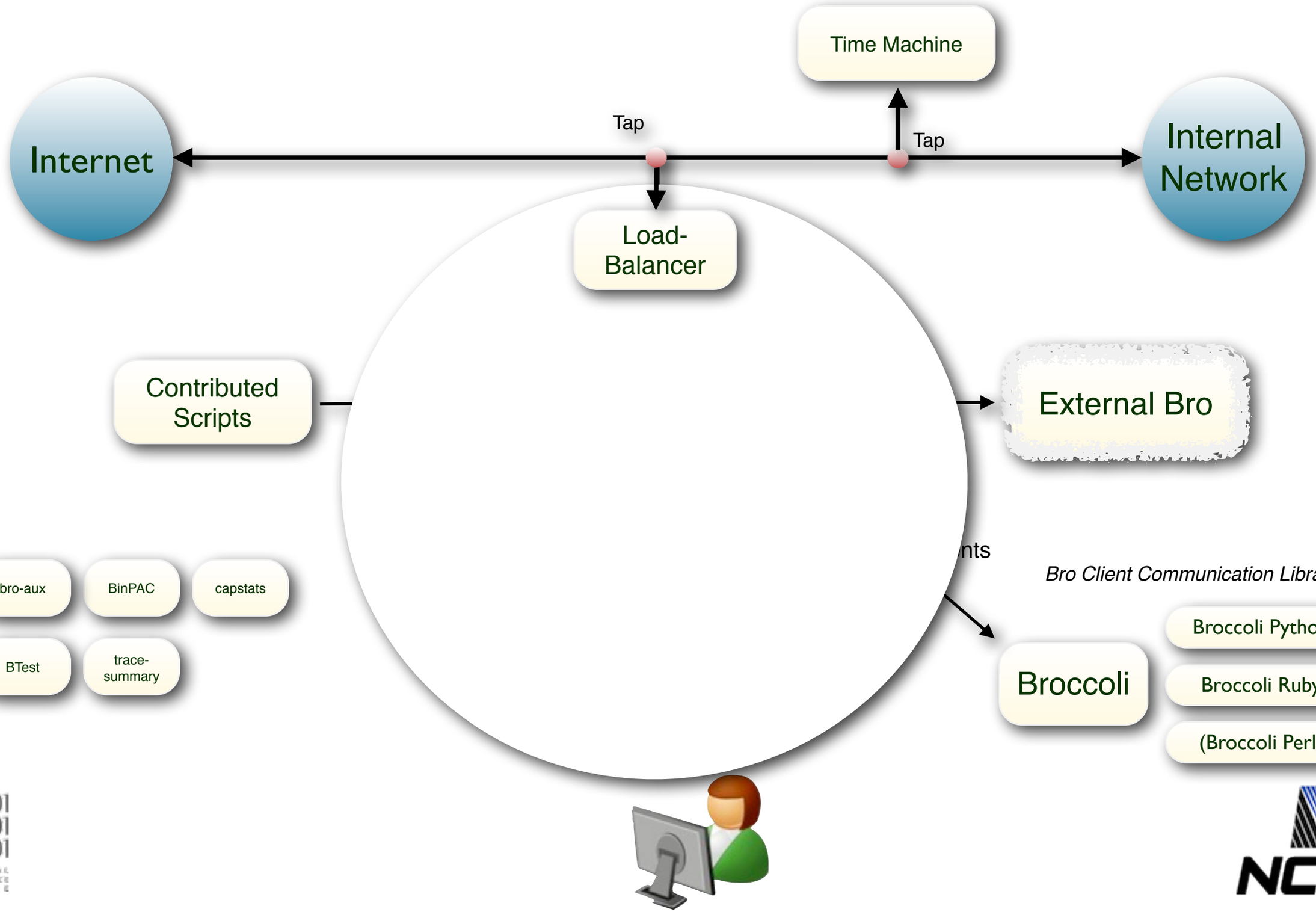
Bro *Cluster* Ecosystem



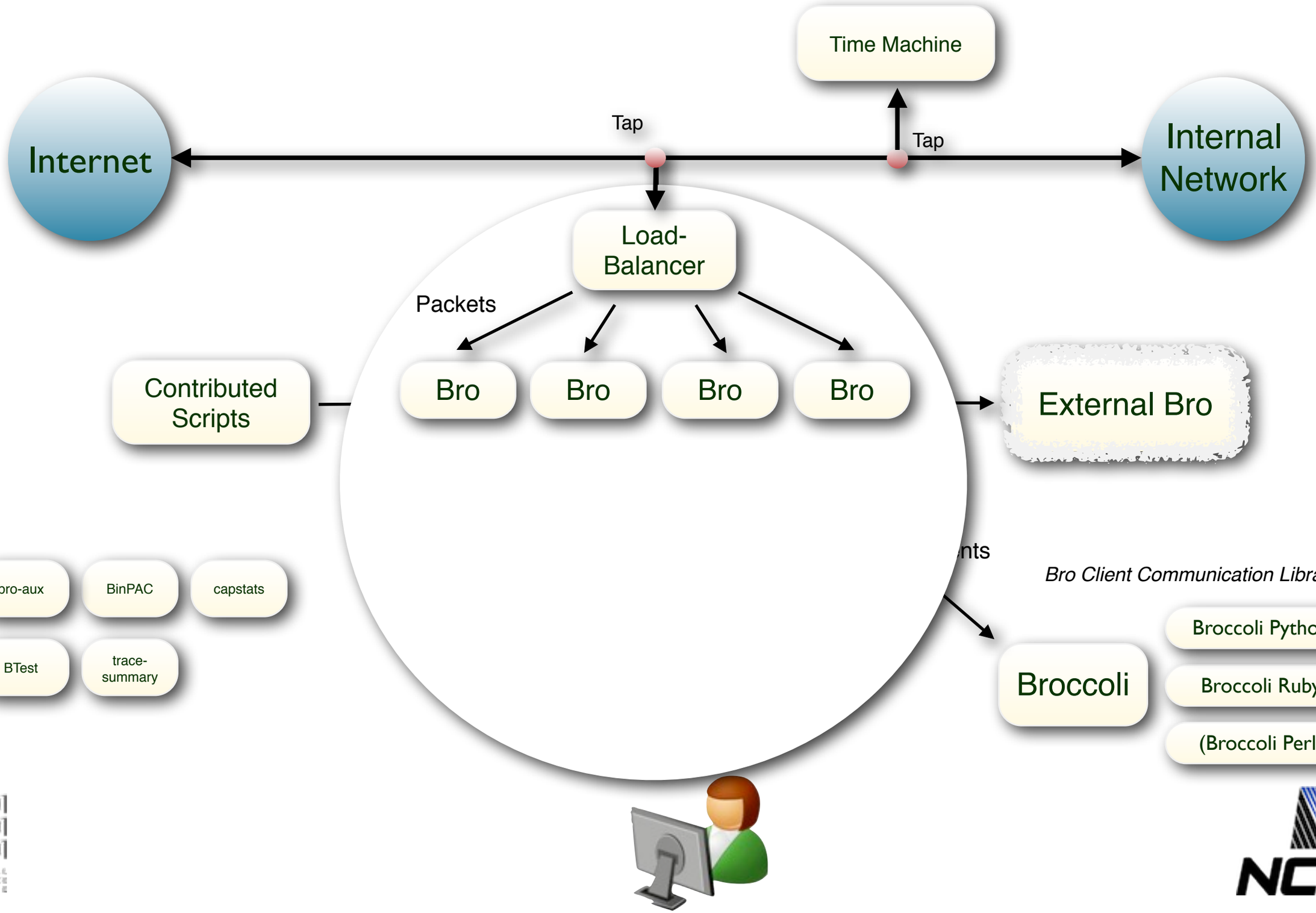
Bro *Cluster* Ecosystem



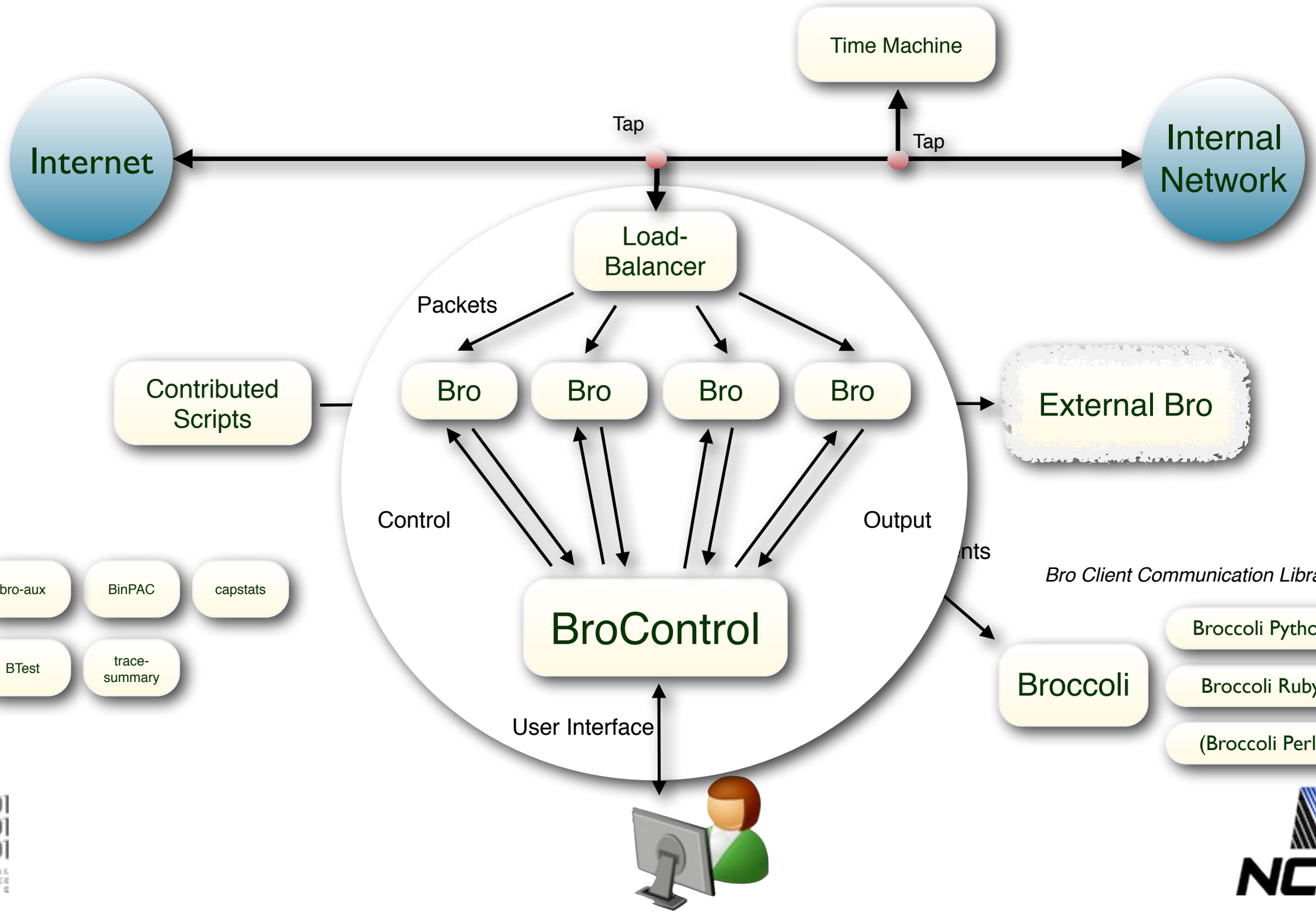
Bro *Cluster* Ecosystem



Bro *Cluster* Ecosystem



Bro *Cluster* Ecosystem



Bro *Cluster* Ecosystem