Stockade Architecture

Introduction

This document contains an architectural overview of the Stockade system, intended to assist those who wish to make modifications to the system.

The Stockade system may be viewed as being composed of three modules:

The historical rehabilitating model (HRM)
Track blacklisted IP ranges and supervise their rehabilitation.

The connection filter
Perform system level filtering of connection attempts, either by querying the HRM, or by observing changes in the HRM and maintaining its own state.

The spam classification client interface
Accept messages from third party spam classifiers (for example SpamAssassin, or an Internet blocking list), populating the HRM with this information.
Historical Rehabilitating Model

Source: CRehabilitatingBlacklist.h

Within the Historical Rehabilitating Model lies the implementation of the rehabilitation process and the logic for managing blacklist contents.

Rehabilitation of blacklist entries is performed either as the result of a blacklist query (via CRehabilitatingBlacklist::connDropCheck), or through the action of the rehabilitation heartbeat (CRehabilitatingBlacklist::rehabHeartbeat).

Heartbeat based rehabilitation keeps a cursor into the blacklist, and iterates through a configurable number of blacklist entries each beat, performing rehabilitation on each one. The heartbeat process also performs aggregation of adjacent blacklist entries - at present, this only serves to reduce the space requirements of the blacklist model, but in the future this may be extended to support a more aggressive neighbourhood poisoning model.
source: CPacketFilter.h

The Stockade 0.2 distribution supports two packet filtering implementations:

- divert socket based filtering, and
- firewall rule based filtering using FreeBSD's IPFW.

The divert socket based implementation is relatively straightforward; once the CDivertSocketFilter::run method is called, the process waits for packets to be diverted by the kernel, querying the HRM to determine if packets should be filtered or allowed to propagate up the TCP stack.

Firewall rule based filtering introduces the challenge that the firewall maintenance module must be kept aware of changes in the HRM state. This is accomplished via the CRuleBasedFilter abstract base class, which registers itself as an observer of the blacklist upon invocation of CRuleBasedFilter::run.

The rule based filter passes events received from the blacklist to an encapsulated firewall maintainer object. Distributed with Stockade 0.2 are two classes that maintain a FreeBSD IPFW based firewall:

- a vanilla IPFW rule maintainer, which communicates with the FreeBSD IPFW
subsystem via system calls, and
- an IPC based firewall maintainer that runs the previously mentioned IPFW rule maintainer within a child process, so that the parent blacklist server process may drop root privileges while still allowing ipfw rules to be modified.

**Spam Classification Client Interface**

```
CClientConn
+handleConnection(stream:CTcpStream)
+registers spam with
+maintains blacklist

CREhabilitatingBlacklist
+registerSpam(ipRange:CIpRange,metric:double)
```

source: CClientConn.h

Communication with third party spam classification software is accomplished via the spam classification client interface. Once mail is received by an smtp server, it may be routed to spam classification software either through a mail server plugin system, or via an end user interface such as procmail.

The result of processing by such a system (which may, for example, be a content analyser) can be communicated to the blacklist server through a TCP transaction. The protocol for this transaction is documented in doc/protocol.txt in the Stockade distribution tarball.