Accept Filters

Mike Silbersack

Accept Filters: Rationale

- Accept filters serve to reduce the number of unnecessary wakeups a typical server will see.
  - Sockets are not passed to the application via accept() until the accept filter's criteria has been met.
- Accept filters shipped with FreeBSD:
  - accf_data: Any data is received on the socket.
  - accf_http: A http HEAD or GET request has been received on the socket.
    - If a string not starting with G or H is detected, the socket is immediately handed to the application.
- Other accept filter ideas:
  - A filter that emits a banner?
  - Would an acceptandread() syscall be useful?

Accept Filters: The API

```
struct accept_filter {
        char    accf_name[16];
        void    (*accf_callback)(struct socket *so, void *arg, int waitflag);
        void *  (*accf_create)(struct socket *so, char *arg);
        void    (*accf_destroy)(struct socket *so);
        SLIST_ENTRY(accept_filter) accf_next;   /* next on the list */
};
```

- accf_data and accf_http use only accf_callback. accf_create and accf_destroy may not be well tested.
- Filters can be dynamically loaded. Dynamic unload can be enabled, but it will probably break your system.

Accept filters: uipc_accf.c

- Handles loading and unloading of accept filter modules, enabling and disabling of accept filters per socket.
- Note: accept_filter_mtx. This is used only to synchronize loading and unloading of filter modules, not for any runtime locking of accept filters.

Accept Filters: hooks into the socket layer

- uipc_socket.c:
  - sosetopt and sogetopt allow programs to set / get the status of accept filtering on a socket. sodealloc removes accept filters during socket destruction.
  - soisconnected is called when the 3WHS completes
    - If no accept filtering, the socket is moved from so_incomp to so_comp
    - If accept filtering is enabled, the socket continues to live on so_incomp. so_upcall and SB_UPCALL are set for that socket. so_upcall is called.
      - Keeping the socket on so_incomp is probably a mistake, due to how we handle incomp overflows in sonewconn. Overflows due to waiting on accept filtering should probably be kicked up to the application rather than dropped.
  - sowakeup will call so_upcall when new data arrives if SB_UPCALL is set.

Accept Filters: accf_data.c

```
static void
sohasdata(struct socket *so, void *arg, int waitflag)
{

        if (!soreadable(so))
                return;

        so->so_upcall = NULL;
        so->so_rcv.sb_flags &= ~SB_UPCALL;
        soisconnected(so);
        return;
}
```

- Simple, yet effective.

Accept Filters: accf_http.c

- sohashttpget handles parsing the incoming data.
  - Early exit if the string does not start with G or H
  - No timeout mechanism, some IIS worms do not trigger the completion criteria and get stuck in here.