TaoSecurity

Dedicated to FreeBSD, network security monitoring, incident response, and network forensics. Email taosecurity at gmail dot com.

Wednesday, March 08, 2006

Improved Bridging for Monitoring in FreeBSD

FreeBSD developer Christian S.J. Peron wrote to me about two commits that improve support for bonding interfaces for use with network taps. He writes:

Let's say that you have a GigE copper tap, and you have the two monitor cables coming into the FreeBSD network analyzer on interfaces em0 and em1. You can aggregate the two links into one logical bridge interface to monitor them:

```
ifconfig bridge0 create
ifconfig bridge0 addm em0 addm em1 up
tcpdump -i bridge0
```

This basically turns em0 and em1 into switch ports. If you want to use this bridge specifically to aggregate one or more network interfaces and pass the packets off to BPF and return, then you can turn off the bridging functionality.

```
ifconfig bridge0 monitor
```

This prevents the bridge code from looking up which port a certain hardware address is attached to, or broadcasting packets out all ports in the event it doesn't know. Essentially, it short circuits the bridging code, which saves a number of mutex acquisitions, list traversals, reducing the load.

We have done this in places which use firewall clusters, I.E. 2 or 3 different PIX firewalls running VRRP

```
ifconfig bridge0 create
ifconfig bridge0 addm em0 addm em1 addm em2 addm em3 addm em4 ad

snort -i bridge0
```

This way, snort works regardless of which firewall has failed over. The bridge is in monitor mode, so it's not actually trying to TX packets out the other interfaces, it just passes the packets it receives to BPF and returns.

This is neat. We won't see it in FreeBSD 6.1, but probably 6.2. Before 6.2, these features will appear in STABLE.

Blog Archive