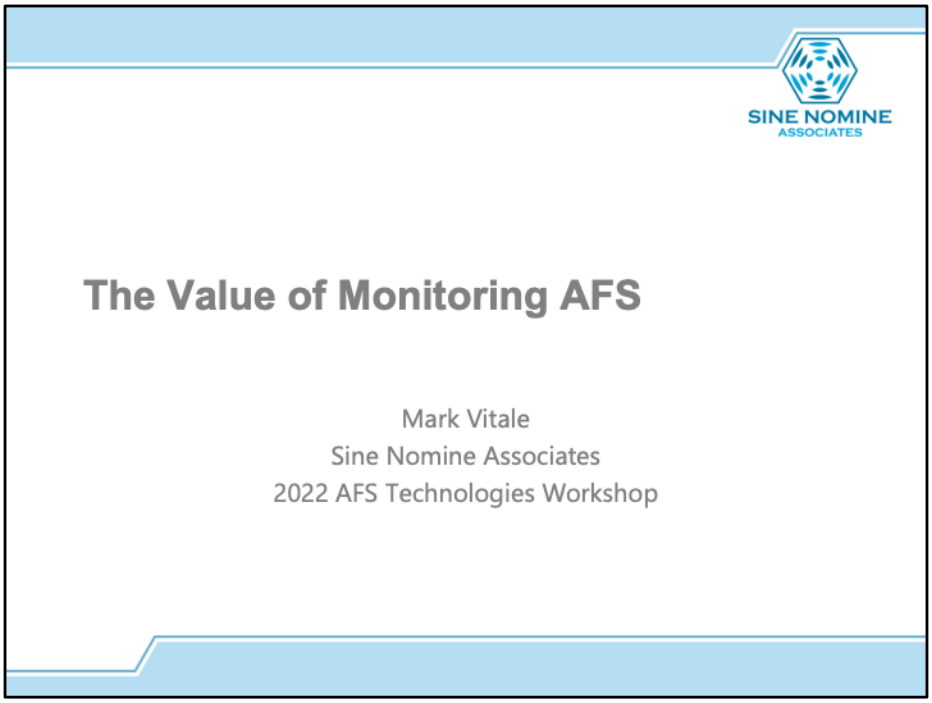


Welcome, and thanks for participating in the conference today!

Today I will be speaking briefly on The Value of Monitoring AFS

The slides will be available after the talk

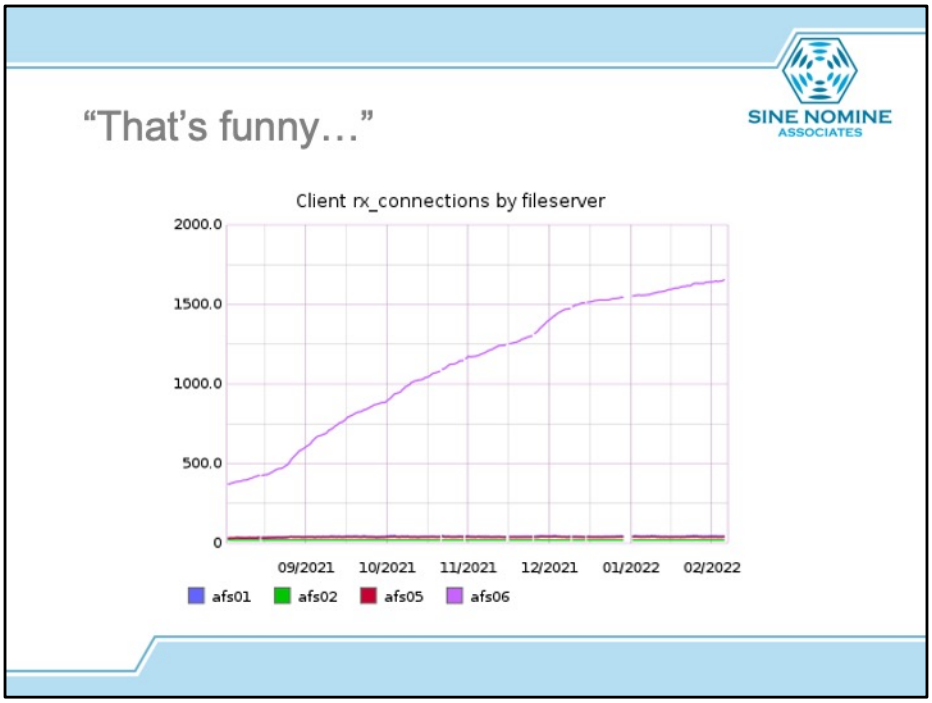
As some of you know, I normally welcome questions and interruptions during my talks
However, In the interest of staying on schedule today,
please defer your questions until the end

Introduce myself briefly

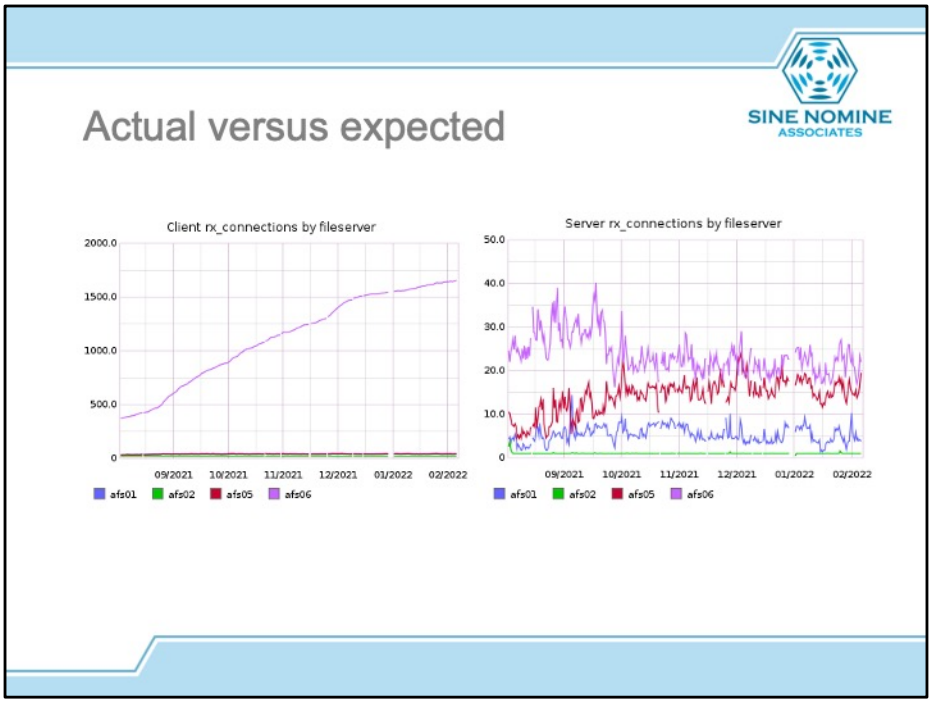
Introduce the talk:
The values of monitoring are well known:
- don't want to preach to the choir
- post mortem troubleshooting
- real time debugging
- capacity planning

- service level agreements
- and more

- but every once in a while, another value of monitoring is finding something unexpected…



this is SNA, not the original site…



note difference in y-axis scale,
constant variation on left is swamped by the magnitude of the leak

Backstory

- A site reported intermittent-but-severe fileserver performance issues
- Root cause:
  - fileserver callback space exhaustion ("GotSomeSpaces")
- Mitigation:
  - Site doubled callback option for all fileservers from `-cb 3000000` to `-cb 6000000`
  - SNA began proactive monitoring of site's callback demand

Metric source

```
$ xstat_fs_test afs06 -collID 3 -onceonly

Starting up the xstat_fs service, one-shot operation

-----------------------------------------------------------
AFS_XSTATSCOLL_CBSTATS (coll 3) for FS afs06.sinenomine.net
[Probe 1, Sun Jun 12 21:54:31 2022]

         1301061 DeleteFiles
        17692775 DeleteCallBacks
         6449457 BreakCallBacks
        59816291 AddCallBack
               0 GotSomeSpaces
           10073 DeleteAllCallBacks
             307 nFEs
             316 nCBs
          512000 nblks
         6979609 CBsTimedOut
               0 nbreakers
               0 GSS1
               0 GSS2
               0 GSS3
               0 GSS4
               0 GSS5
```

GotSomeSpaces is triggered when nCBs or nFEs reaches the value of nblks (fileserver option –cb <nnn>)
- thus as nCBs / nFEs approaches the value of nblks, the risk of GotSomeSpaces increases

GSS1-5 are just a developer-only breakout of the GotSomeSpaces count

Metric source (modified)

(https://gerrit.openafs.org/#/c/14359)

```
$ xstat_fs_test afs06 -collID 3 -onceonly -format dsv
1655085988,1,afs06.sinenomine.net,3,DeleteFiles,1301061
1655085988,1,afs06.sinenomine.net,3,DeleteCallBacks,17692775
1655085988,1,afs06.sinenomine.net,3,BreakCallBacks,6449457
1655085988,1,afs06.sinenomine.net,3,AddCallBack,59816526
1655085988,1,afs06.sinenomine.net,3,GotSomeSpaces,0
1655085988,1,afs06.sinenomine.net,3,DeleteAllCallBacks,10073
1655085988,1,afs06.sinenomine.net,3,nFEs,307
1655085988,1,afs06.sinenomine.net,3,nCBs,314
1655085988,1,afs06.sinenomine.net,3,nblks,512000
1655085988,1,afs06.sinenomine.net,3,CBsTimedOut,6979611
1655085988,1,afs06.sinenomine.net,3,nbreakers,0
1655085988,1,afs06.sinenomine.net,3,GSS1,0
1655085988,1,afs06.sinenomine.net,3,GSS2,0
1655085988,1,afs06.sinenomine.net,3,GSS3,0
1655085988,1,afs06.sinenomine.net,3,GSS4,0
1655085988,1,afs06.sinenomine.net,3,GSS5,0
```

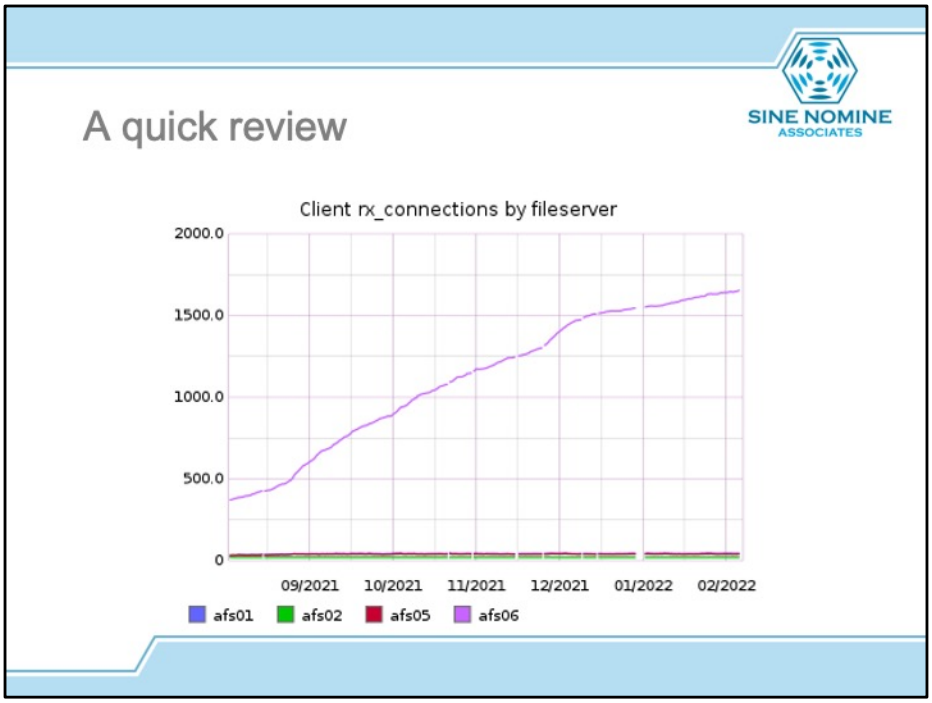
Search for other warning indicators

SINE NOMINE ASSOCIATES

- SNA created additional metric charts
  - to set a baseline for normal behavior
  - to look for load patterns that may precede increased callback load
- Client rx_connection chart revealed an unexpected resource leak

GotSomeSpaces may be provoked by underconfiguration, -OR- a misbehaving client or application:

- e.g. find (walking the AFS RW space)



to refresh your memory from the first slide …

Metric source

```
$ rxdebug afs06 7000 -rxstat -noconn
Trying 207.89.43.113 (port 7000):
Free packets: 1496/4727, packet reclaims: 30034, calls: 31474076, used FDs: 64
not waiting for packets.
0 calls waiting for a thread
130 threads are idle
0 calls have waited for a thread
rx stats: free packets 1496, allocs 877522263, alloc-failures(rcv 0/0,send 0/0,ack 0)
   greedy 0, bogusReads 2 (last from host 268a4b6a), noPackets 4176472, noBuffers 0, selects
0, sendSelects 0
   packets read: data 3290780073 ack 917352509 busy 254 abort 8351 ackall 0 challenge 10955
response 504936 debug 1225006 params 0 unused 0 unused 0 unused 0 version 0
   other read counters: data 3290779614, ack 917258590, dup 13741011 spurious 93305 dally 439
   packets sent: data 1662593046 ack 1709778646 busy 62 abort 272781 ackall 0 challenge
518028 response 10955 debug 0 params 0 unused 0 unused 0 unused 0 version 0
   other send counters: ack 1709778646, data 1662345898 (not resends), resends 247148, pushed
0, acked&ignored 1373122790
       (these should be small) sendFailed 24304, fatalErrors 105
   Average rtt is 0.001, with 1560512427 samples
   Minimum rtt is 0.000, maximum is 26.154
   31 server connections, 2080 client connections, 322 peer structs, 87 call structs, 78 free
call structs
   0 clock updates
```

Metric source (modified)

(https://gerrit.openafs.org/#/c/14358)

```
$ rxdebug afs06 7000 -rxstat -noconn -raw
nFreePackets 1487
nPackets 4727
packetReclaims 30034
callsExecuted 31474425
usedFDs 64
waitingForPackets 0
nWaiting 0
idleThreads 130
nWaited 0
freePackets 1487
. . .
minRtt 0.000
maxRtt 26.154
nServerConns 49
nClientConns 2080
nPeerStructs 322
nCallStructs 87
nFreeCallStructs 63
```

Known call/conn issues

commit	OpenAFS components	master	1.9.x	1.8.x	1.6.x
rx: prevent leaked rx_call structures	all	n/a	n/a	n/a	13693
rx: fix call refcount leak in error case	all	12781	12781 1.9.0	12783 1.8.0	n/a
rx: prevent leakage of non-cached rx_connections	all pthreaded ubik clients	13042	13042 1.9.0	14514 1.8.8	needed
afs: Avoid NatPing event on all connection - not a true leak, but other leaks may exacerbate its effects	Unix CM only	14312	14312 1.9.0	14364 1.8.8	n/a
afs: Avoid creating unused conns - not a true leak; superfluous conns are cleaned up after NOTOKTIMEOUT 2h, then another set pops up within 10m (checkservers up) - unlikely to be backported to 1.8.x or 1.6.x	Unix CM only	14637	pending	needed	needed
rx: prevent leak of cache manager NAT ping rx_connections	Unix & Windows CMa only	14951	needed	needed	n/a
rx: prevent leak of client rx_connections - possibly more likely in 1.8.x and up	all	14952	needed	needed	needed

Gerrit number color key:
- green merged
- yellow under review

The leak didn't match any known leaks up to that point

The last one is the one that triggered this talk – it is a long-standing problem – a race between the Rx listener and
the next-to-last one was discovered while investigating and testing the last one

AFS monitoring: a simple start

SINE NOMINE ASSOCIATES

- Many useful metrics available without config changes or authentication:
  - `rxdebug <host> <port> –rxstat –noconn`
  - `xstat_fs_test <fileserver> -collID 2 3`
- Send metrics to existing system, e.g.:
  - carbon/graphite
  - collectd/grafana
  - Splunk

carbon/graphite are quite long in the tooth
I only used them because they were easy to setup
I'm a developer, not an admin!

solicit suggestions for Q&A at the end

OpenAFS suggested key metrics

SINE NOMINE ASSOCIATES

component	good	better	source
-all-	threads have waited	idle threads	`rxdebug -rxstat`
fileserver	GotSomeSpaces	nCBs / nblks	`xstat_fs_test -collID 3`
-all-	per RPC counts & max queue and run time	per RPC avg. queue and run time	`initial setup: -enable_process_stats rxstat_enable_process` `rxstat_get_process`

I'm sure we could think of many more, but the first two are critical

the third one does require minimal configuration –
- -enable_process_stats
- may also be enabled at runtime via rxstat_enable_process

maybe one or two more here?

Conclusion

SINE NOMINE ASSOCIATES

- The value of monitoring AFS cells:
  - characterize what "normal" looks like for your site
  - configure for optimal performance
  - identify abnormal behavior before severe symptoms arise
  - sometimes… you find the unexpected

a benefit to your site as well as the community

thank you

Questions & responses

SINE NOMINE
ASSOCIATES