Software Engineering Conference Russia
October 2017, St. Petersburg

On one source of latency in NFSv4 client

D. Irtegov, P. Belousov, A. Fedosenko, A. Fal,
Novosibirsk State University

How we got interested in NFS latency

1. Joint NSU-Parallels research project to find a storage solution for load-balanced shared web hosting

How we got interested in NFS latency

1. Joint NSU-Parallels research project to find a storage solution for load-balanced shared web hosting
2. Requiremens:
    1. Networked and shared
    2. Mostly (~90%) read, but ~10% write
    3. Average file size ~10-100kb
    4. Low latency
       every web page open requires open(2) or stat(2) of ~100 files

If you want to know how load-balanced shared web hosting translates to these requirements, we can discuss it during Q&A

Available storage types

	iSCSI	NFS	Cluster FS
Shared	-	+	+
Performance (latency)	+	-	+
Stability	+	+	*

* Cluster FS balance between latency and sensitivity to network failures. Essentially this is a general issue of distributed locking.

What we mean by “latency (-)”

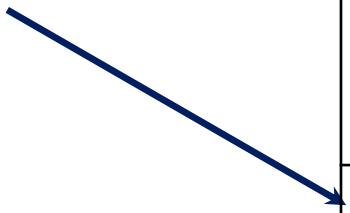
Sustained speed of opening single dynamic web page

- Joomla
- Apache 2.0+mod_php
- Apache Jmeter
- 100 requests (hot cache)
- Milliseconds per request

	average	80%	min
iSCSI	111	100	91
NFS	130	142	91
NFS loop	115	108	92

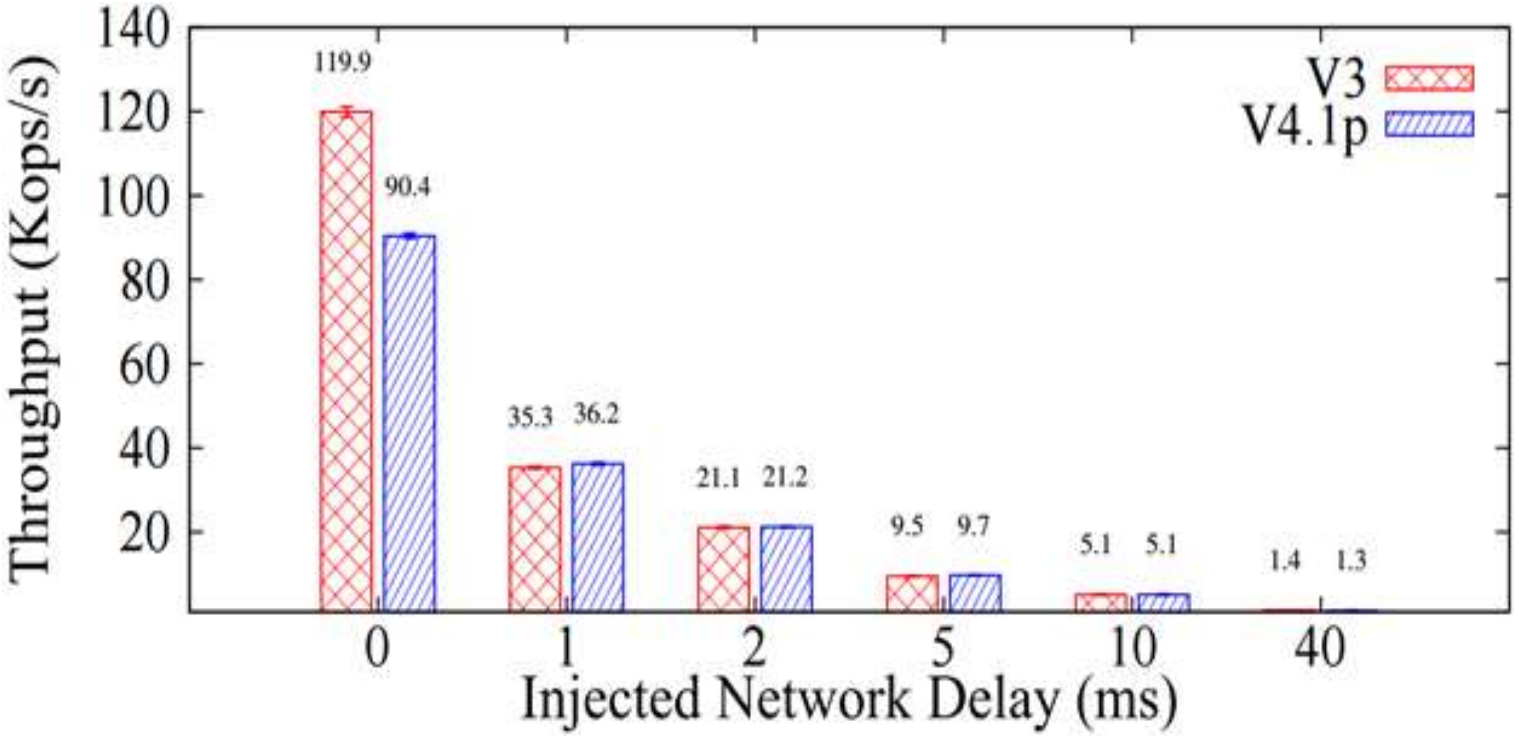
What is NFS loop

Virtual block device
(loopback device)
Located on NFS share



	average	80%	min
iSCSI	111	100	91
NFS	130	142	91
NFS loop	115	108	92

Other research



Source: Ming C., Dean H., Geoff K., Soujanya S., Vasily T., Arun O., Erek Z., Ksenia Z. Linux NFSv4.1 Performance Under a Microscope, Techreport FSL-14-02, Nov 2014

SECR

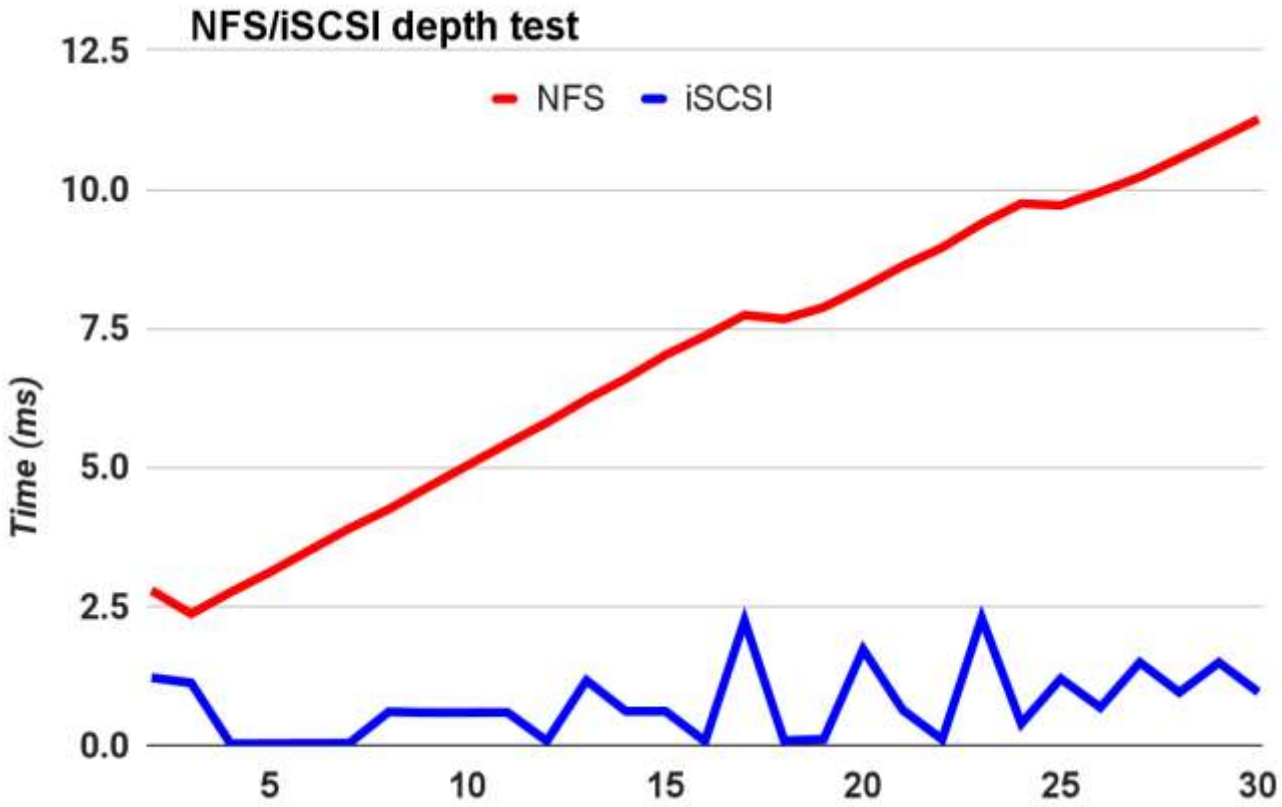
Generally accepted facts and myths

1. NFS (even NFSv4) is “chatty” protocol compared to iSCSI
2. Sensitive to network latency
3. Slow on “Metadata-intensive” operations

   - this is strange because NFS uses high-level RPC while iSCSI and cluster FS read raw metadata
4. This is a distributed locking issue

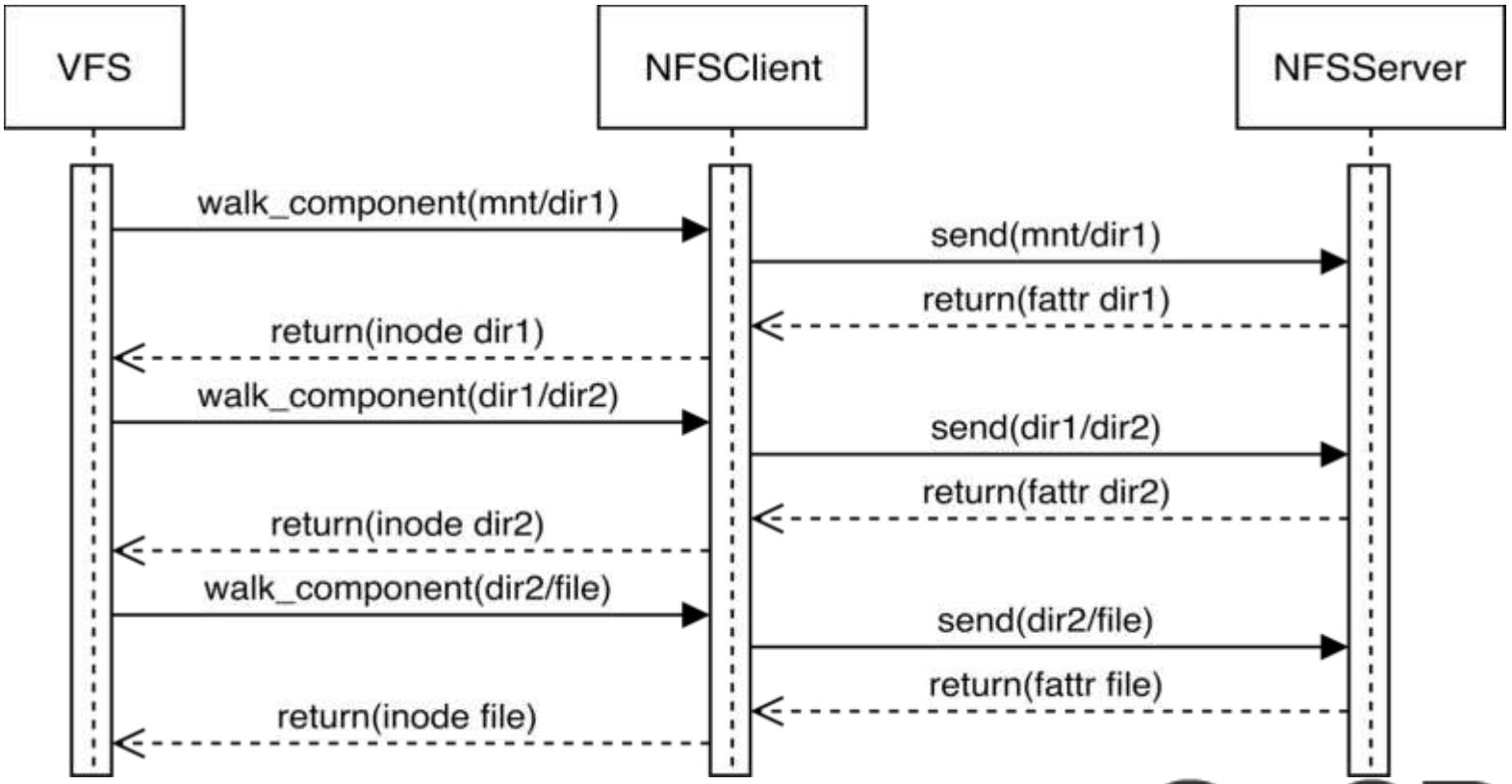
   - spoiler: this is false!

Time of open(2) in nested dirs



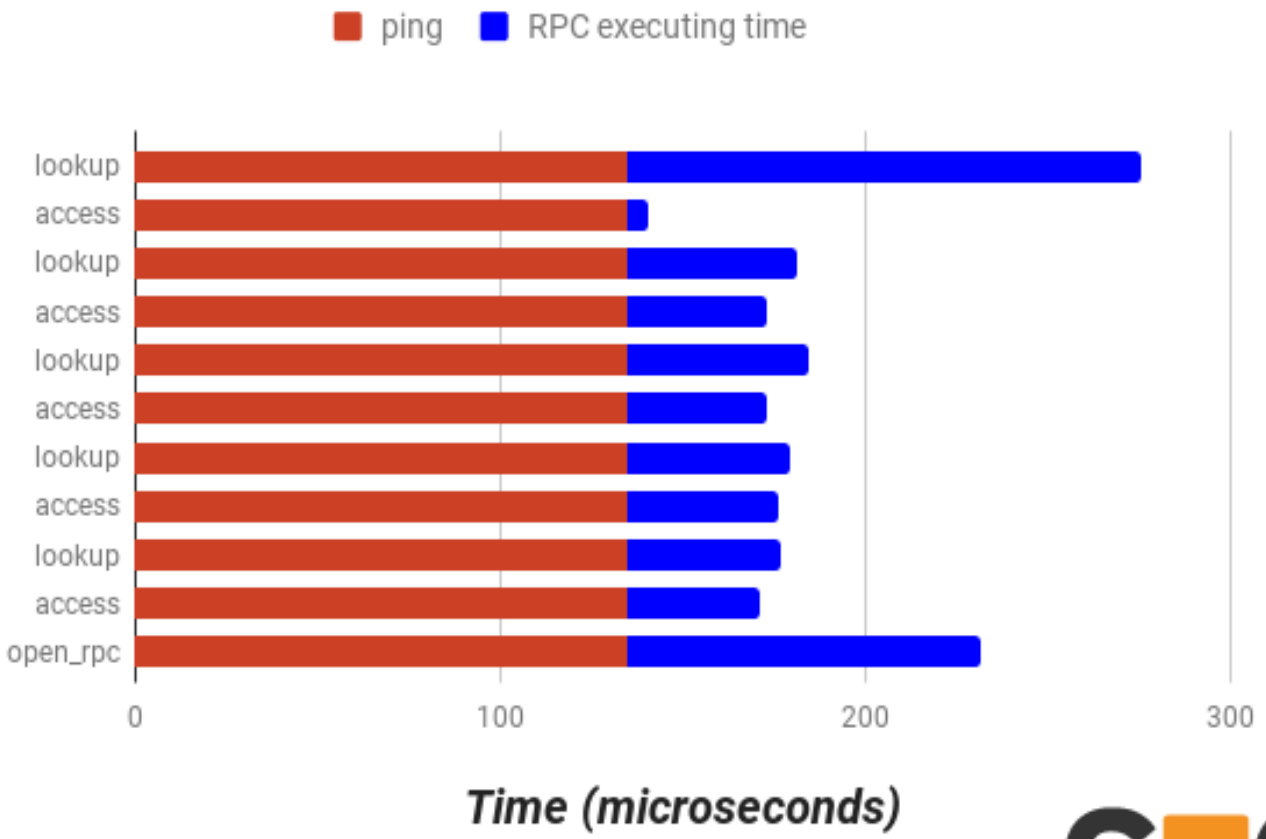
SECR

RPC call sequence (Linux 3.10)



SECR

Why it is important



Compound requests

1. In modern networks, RTT is dominating all other sources of latency
2. Modern protocols (iSCSI, NFSv4, SMB 2, SPDY/HTTP 2.0) have features to combat this: command queuing, compound requests, etc
3. NFS v4 has compound requests and clients use it
4. But not for nested lookups
   see paper for rpcdump of Solaris, FreeBSD and Linux clients

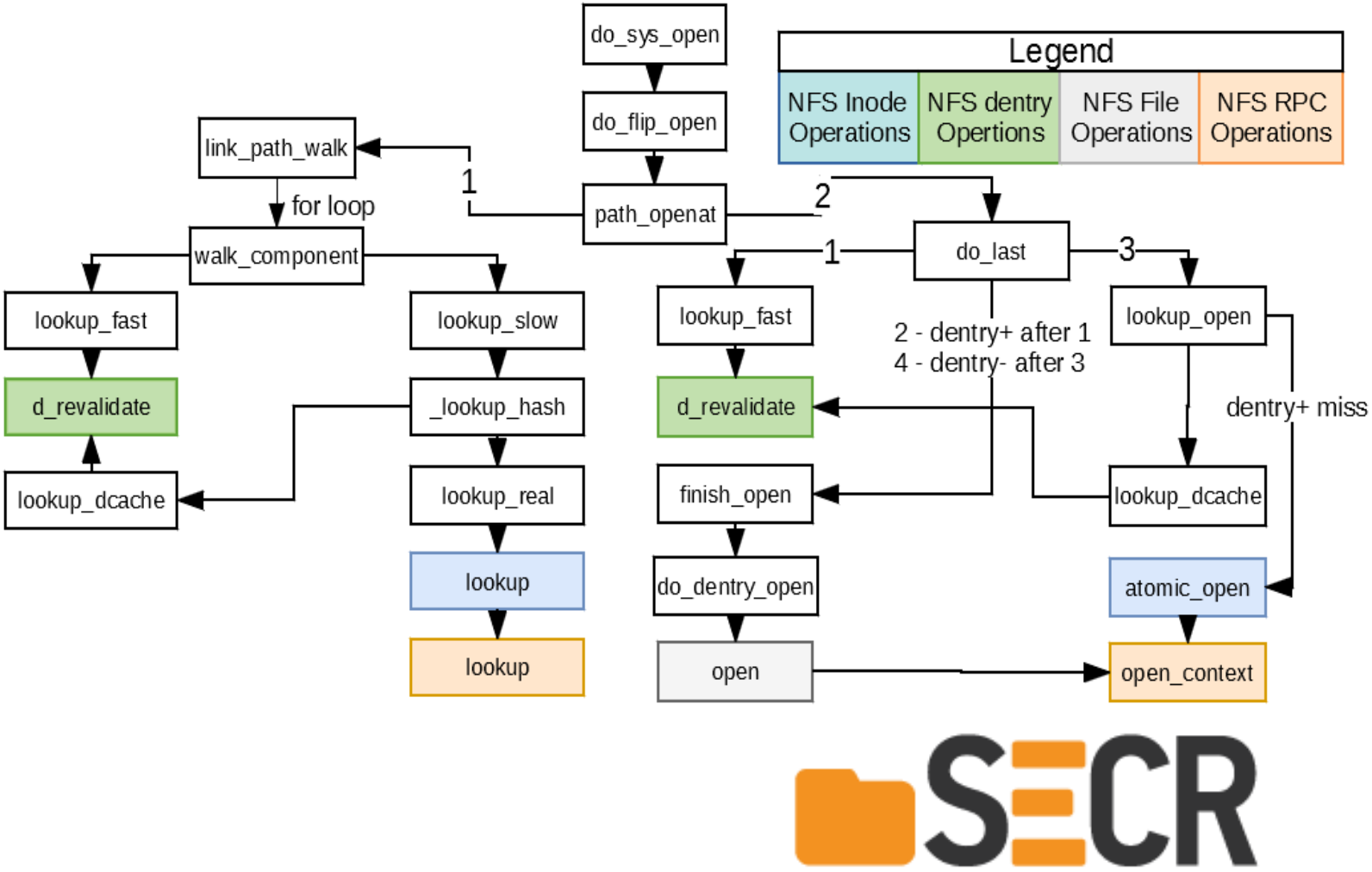
Compound requests (continue)

Chen, M., Hildebrand, D., Nelson, H., Saluja, J., Subramony, A. S. H., & Zadok, E. (2017, February). vNFS: Maximizing NFS Performance with Compounds and Vectorized I/O. FAST

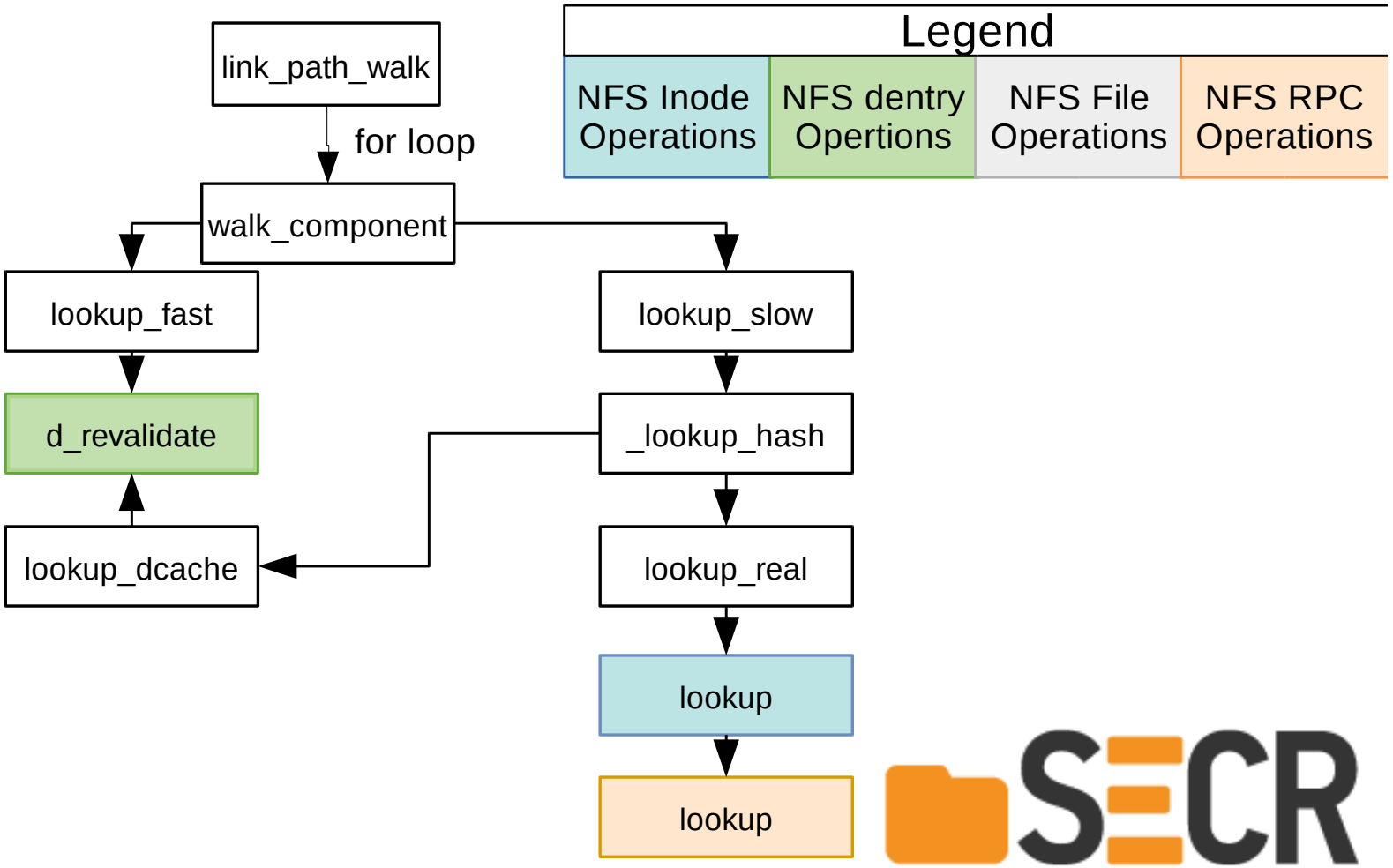
1. Userland NFS client library
2. Big performance boost
3. Servers accept compound lookups
4. They say kernel client cannot do compound lookup due to POSIX semantics

   ????

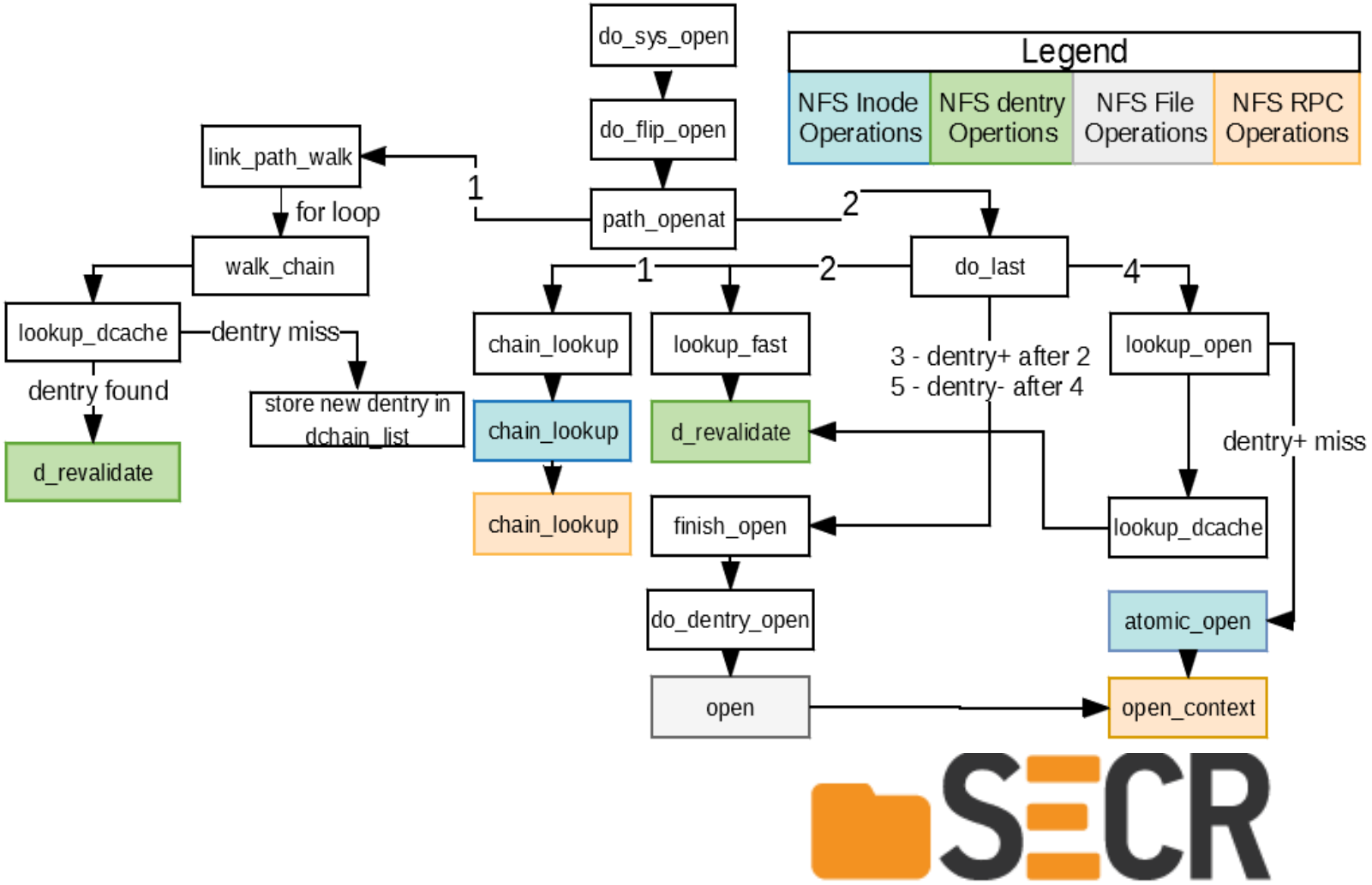
Virtual File System layer



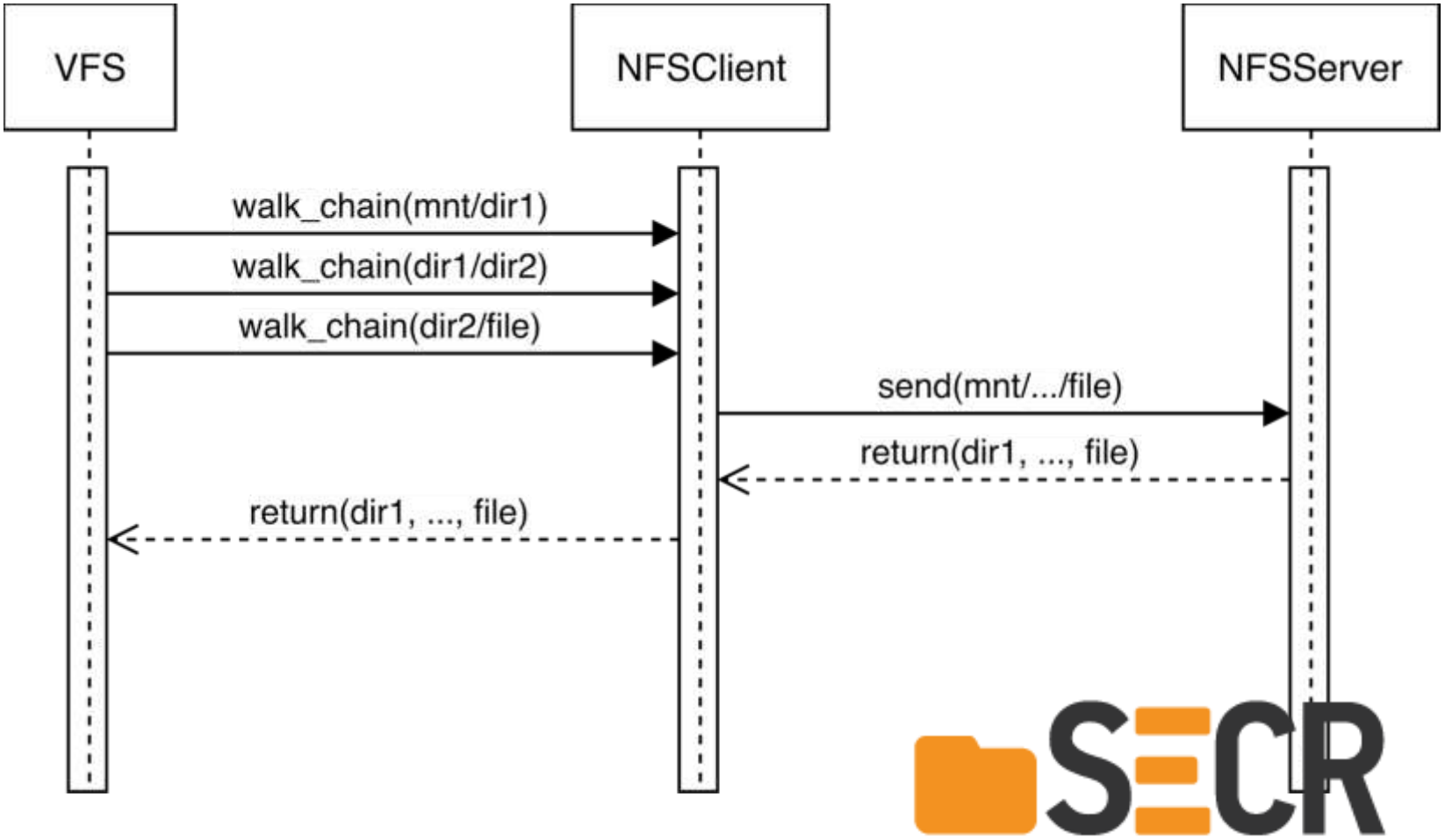
VFS lookup logic (link_path_walk)



Modified VFS



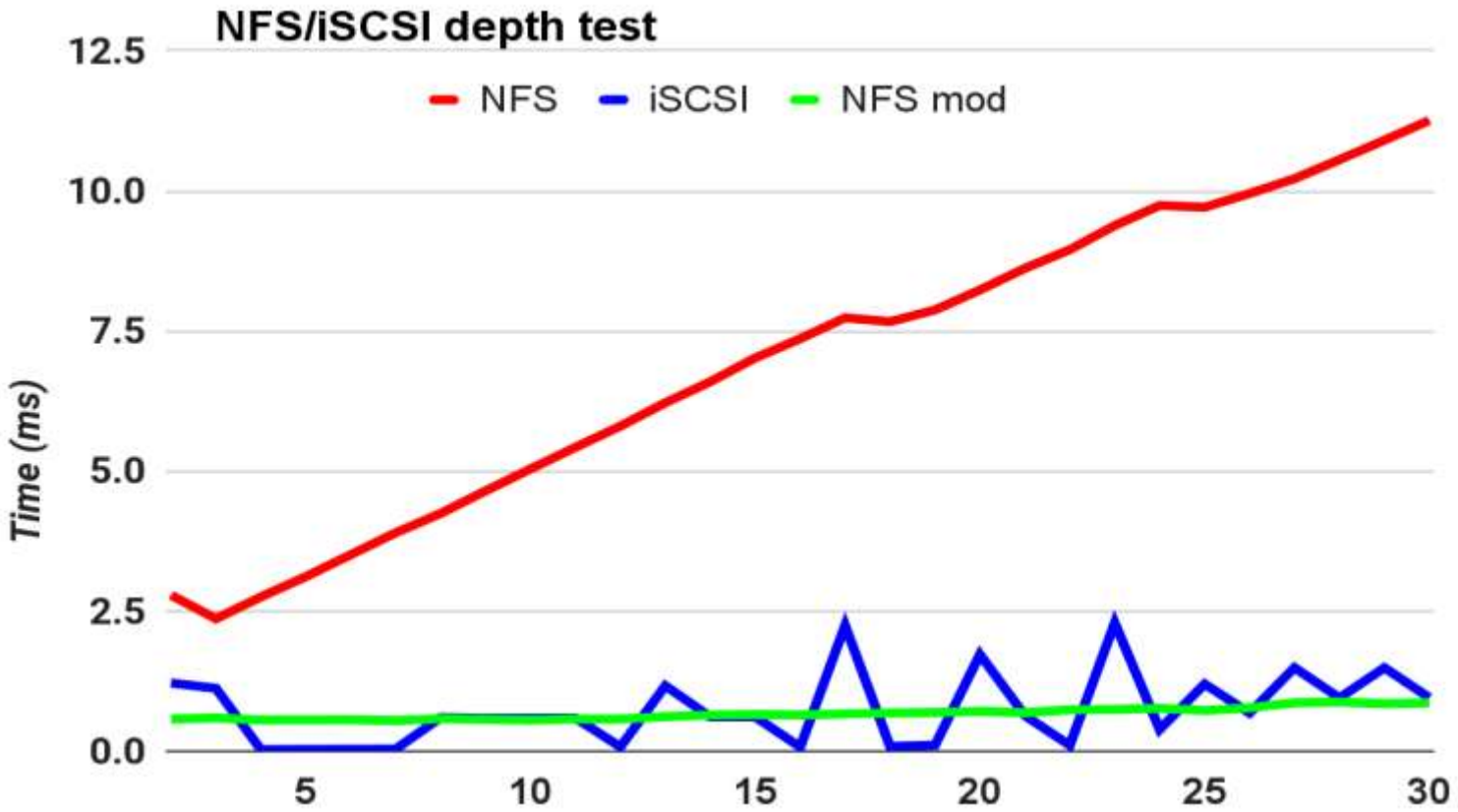
RPC call sequence



Special cases

1. Missing files and directories
   - NFSv4 handles all RPC until first fail
2. Symbolic links
   - NFSv4 cannot do lookup in a symlink
3. “..” directory entries
   - Split pathname to several compounds
4. Server-side limit on compound length
   - Currently not well handled
5. Mount points
   - Locked in VFS cache

Results (hierarchical open(2) test)



SECR

Results (web server test)

	average	80%	min
iSCSI	111	100	91
NFS	130	142	91
NFS loop	115	108	92
NFS mod	108	117	92

Results (where you can find it)

- https://github.com/NSUExplab/nfs4compound
- Patch against CentOS 7 Linux 3.10 kernel
- NOT READY FOR PRODUCTION
- We are open for bug reports and suggestions

DO NOT TRY THIS ON PRODUCTION!

Results (where you can find it)

- https://github.com/NSUExplab/nfs4compound
- Patch against CentOS 7 Linux 3.10 kernel
- NOT READY FOR PRODUCTION
- We are open for bug reports and suggestions
- We plan to have something worthy pushing to mainline kernel probably next year

Q&A?