Are Our Networks Trashing Our Files?

Craig Partridge

January 24, 2020

1 The Problem

There is a distressingly good chance that scientists working with big scientific data sets are working with corrupted data.

A recent study suggests that about 1 file in 121 scientific large data files is delivered by a file transfer tool (FTP, scp, etc.) with undetected errors [2]. In the study, these errors were then discovered by computing a file checksum (e.g. a digital signature check) on the file. The checksum is 32-bits and a naive computation suggests that about 1 in 520 billion (121×2^{32}) transfers will result in a delivered file that is not an accurate copy of the original file. That is iff the file transfer protocol computes a file checksum. About half of all scientific data transfers are done without file checksum.

This is a single study. How plausible are its results? More plausible than we would like. Consider the following points:

- We know that the TCP checksum is quite weak for many types of errors [4].
- We know that historically, most TCP errors are host and router induced errors such as memory problems, bus timing issues, and the like [4].
- There are reasons to believe that middleboxes may be overwriting checksums (e.g. stamping corrupted data with new checksums that suggest the data is good [3]).
- Link level error rates may be going up, due to heavy use of WiFi (whose error rates go up as speeds go up [1]). These errors may be stressing the ability of CRC-32 to catch link errors.
- Anecdotally, scientists are trying hard to protect themselves from these errors. There's a trend towards always trying to copy the authoritative copy of the file (that is, avoiding the use of repos) and copying from multiple sites and comparing the result.

2 What can we do?

The first order problem is that we do not know what kinds of errors are occurring in today's Internet. The last major study was done 20 years ago [4]. The

effectiveness of an error detection and correction scheme depends on the types of errors that are expected. For instance, many checksums and CRCs focus on individual bit errors.

Second, there has been very little study of checksums wider than 32-bits, outside of cryptographic hashes (which are the wrong solution). The fact that we are protecting billions of annual file transfers with a checksum whose range of values is also measured in the (low) billions is an invitation to error.

Once we know the error patterns and we have studied checksums, we are in a position to create modern file transfer protocols for the 21st century.

References

[1] Gábor Fehér. Bit-error analysis in wifi networks based on real measurements. In Róbert Szabó, Hua Zhu, Sándor Imre, and Ranganai Chaparadza, editors, *Access Networks*, pages 127–138, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[2] Zhengchun Liu, Rajkumar Kettimuthu, Ian Foster, and Nageswara S. V. Rao. Cross-geography scientific data transferring trends and behavior. In *Proceedings of the 27th International Symposium on High-Performance Parallel and Distributed Computing*, HPDC '18, pages 267–278, New York, NY, USA, 2018. ACM.

[3] Jan Rüth. Crc and checksums today. personal communication, January 2020.

[4] Jonathan Stone and Craig Partridge. When the crc and tcp checksum disagree. In *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, SIGCOMM '00, page 309–319, New York, NY, USA, 2000. Association for Computing Machinery.