

Saturday, February 6, 2010

## Perceived Risk

Humans have a design flaw, they don't have a sense for risks. When you look at the risks from a more scientific perspective, the probability of dying in a terror attack is vastly lower than the probability of dying because you've committed suicide. With a realistic view to the risks, you should look in the mirror each day and observe yourself, not to other peoples in the aircraft, the subway or somewhere else. However the fear-mongers were able to implant us other feelings and so we strip-search people looking differently from us, we even start to strip-search everybody by technology like backscatter scanners.

Why do i talk about this? Well ... there is an technology in ZFS, that allows you to work with probabilities to speed up things. It's the deduplication part. You can use it in a way, that doesn't verify bit-for-bit if a block is really identical to another before just storing a pointer to the already stored block.

In this non-verifying mode, the system just decides on the foundation of the checksums of both blocks. If the checksum of two blocks is equal, the blocks are considered as equal and the system doesn't check it bit-wise. So you can save a read-IOP.

Most people in IT know about something called hash collision. Such collisions are in the nature of hashing. It's in the nature of hashing, that when you you sort a large amount of things in a smaller amount of buckets, that you will end with more than one thing per bucket. It's even that way, that a perfect mechanism would end with absolutely the same numbers of things in each bucket.

That said, it sounds totally unreasonable just to rely on hashes to compare blocks. Many people talk about something called birthday problem, when you start to talk about hash-based de-duplication. The Wikipedia defines it as following: In probability theory, the birthday problem, or birthday paradox pertains to the probability that in a set of randomly chosen people some pair of them will have the same birthday. In a group of at least 23 randomly chosen people, there is more than 50% probability that some pair of them will have the same birthday. Such a result is counter-intuitive to many. What birthdays are to humans, hashes are to our blocks. Many people think about this high probability, when they think about the deduplication. But you can just have birthday on 365 days, not  $2^{256}$  days.

But there is an interesting table on the Wikipedia-Page as well: When you want to see a hash collision with probability of of a hash collision, you have to hash blocks. That are roughly blocks, translating 316 octillion, 912 septillion, 650 sextillion, 57 quintillion, 57 quadrillion, 350 trillion, 374 billion, 175 million, 801 thousand and 344 blocks. Given you are using 128kB blocks this gives you a storage capacity of yottabytes. Just a comparison: That's roughly times the human knowledge (i love Wolfram Alpha for such comparisons).

Why did i choose a  $1 \times 10^{-18}$  and why is there a column in the wikipedia page. Well ... it's the same probability that you see a unrecoverable bit error from your favorite high-end hard disk.

So when you have stored 316 octillion, 912 septillion, 650 sextillion, 57 quintillion, 57 quadrillion, 350 trillion, 374 billion, 175 million, 801 thousand and 344 blocks to have a probability of a false positive duplicate roughly as high as the probability of having a uncorrectable bit error from you hard disk.

For real world dataset sizes at a customers the probability of a false-positive deduplication doesn't look as a problem you should worry about.

However everyone which i talked to about deduplication still thinks that she or he needs this verify read, that's unacceptable to do it without it. But's let's face it: The probability of an undetected error of you ECC ram is much higher, the error of an undetected error while reading data from your disks is much higher. Even the probability of getting a undetected erroneous packet via network is much larger. At the end Ethernet frames are just protected by CRC32. The checksum mechanism protecting the payload of packets in IPv4 is even much more prone to undetected errors than CRC2.

However many people still rely on filesystems without checksums for they data. The same people doesn't make IPsec mandatory to protect the data while on transport (The sec in IPsec is not only about confidentiality. It's about integrity, too). The same people use x86 servers without ECC everywhere. The same people work with TCP Checksum Offload, thus computing the checksum in the networking card allowing the packet to be unprotected from the CPU to the NIC -

## Blog Export: c0t0d0s0.org, http://www.c0t0d0s0.org/

and there are a lot of components between both.

So ... at the end it's really strange that people have problems with a checksum-only deduplication, but don't have problems with mechanisms that put their data at a much larger risk. As i wrote before: Humans don't have a natural real sense for risks and aren't trained to assess risks correctly. So they have fears about the consequences, where they shouldn't have them, and they take risks, they shouldn't take. At the end it's about a perceived situation, and this is the reason why there is a verify option in ZFS deduplication.

Posted by Joerg Moellenkamp in English, Solaris, Sun/Oracle at 14:41

Would you please remove "islamistic" from terror attack! I am an Egyptian muslim, and an avid reader of your blog. I was very offended. Simply and automatically linking Islamic religion to terror actions, is just unacceptable. I would really have expected a more informed opinion from someone of your intelligence.

If you do decide to remove the word, respecting a wide range of your readers base. Please do not publish this comment either  
Anonymous on Feb 6 2010, 15:39

You got this part of article totally wrong. After 9/11 many people thought that there is an extremely high probability of being killed by a terrorist plot, they were even sure that it would be one driven by Al-Quaida. Fear-mongers with their own interests were able to implant this into the minds of western people. And it still occurs ... every know and then a politician or official tries to tell us "High risk of a terror done by islamistic fundamentalists. We need more power for the police"

From the perspective on many people, the perceived risk of being killed by an islamistic terror plot, was high, when it was in reality zero for all practical purposes. However the probability of being killed in a car crash is 1:10000, however nobody tells us not to drive, the probability of being killed by consuming alcohol is higher than being killed by a terrorist. But alcohol isn't prohibited. The risk of a death because of lung cancer is vastly higher than dying by the hand of the terrorist. But you can still buy cigarettes in any supermarket. The risk of being killed by falling unluckily due to the ice on northern germany streets is much higher at the moment than dying due a terrorists hand. However we have extremely expensive anti-terror security measures (and they will be more expensive, when our politician get their will of buying backscatter scanners) but we have almost no salt left in our storehouses to remove the ice on our streets (they even stopped winter service on autobahnen). It's more probable to be killed by your doctor because of a faulty diagnose, there are numbers from a study in 2000 that 96.000 people die due to errors of their doctors, so even in 2001 doctors were 32 times more effective than Al-Qaida and their islamistic terror , it's more probable due to bacterias in the food, due to an accident at work (deadly accidents at work in 2008 in germany: 1046, deaths due to islamistic terror in 2008 in germany: 0).

Despite of the fact that almost every way to die (expect being hit by a meteorite right in the head) is more probable than being killed by a islamistic terrorist, i know from a distant friend born in Iran (he and his parents left Iran when he was roughly a year) that he has always problems with traveling, he is sure that his luggage is checked more intensive (and in the mean time numerous people purchase alcohol/cigarettes in the duty-free shop, by the way). A colleague of my brother told my brother, that his father (an older business man from hamburg) isn't allowed to travel in the US, assumingly for no other reason than being born in middle east.

It's not about insulting muslims, it's about showing how skewed the risk perception of people is, it's about how idiotic it is to believe, to die due to a terror attack, while not thinking over much more probable risks.

However as i hold the muslim belief in high regard, i changed the part of this article. The comment was automatically published. As long none of the automatic systems recognize it as spam, it appears on the website instantaneous.  
Anonymous on Feb 6 2010, 16:34

The math is spot-on but assumes that the blocks are chosen uniformly at random. We've already seen construction of small documents that collide under MD5, and SHA-1 isn't looking so hot (though is not yet broken, and, in fairness, AFAIK, the SHA-2 family currently has a grand security margin). The Fletcher checksums used by default in ZFS are not intended to be cryptographically strong, so it should be quite reasonable to carry out collision attacks.

IOW, if your ZFS store is writable by an adversary who can gain advance knowledge of (the hashes of) blocks to be committed, hash-only-dedup may allow silent corruption of files.

Actually, since Fletcher can't distinguish between blocks of all ones and all zeros, if a dataset is deduped and using Fletcher as its checksum and the first block written is 0xFFFFFFFF..FF, won't any subsequent attempt to write a block of all zeros instead get replaced with one of all ones?

Anonymous on Feb 6 2010, 20:37

Joerg, I agree about the assessment of risk. One's datacenter will probably be soaked by spring floods before one ever encounters such a collision. Even so, maybe there are ways to enjoy dedup=verify, without paying a major performance price.

ZFS checksums are extensible, so I imagine that even more capable SHA-2 algorithms (such as SHA-384 and SHA-512) will appear in ZFS, well before we see operational pools approaching this size. Beyond that, SHA-3 is on the way.

More processor power is required, but newer enterprise systems should offer hardware crypto acceleration, much like the UltraSPARC-T2 has already. And CPU's ought to continue to become generally faster, of course. So the algorithm alone should not compromise performance.

Question: Even if one chooses dedup=verify, could the L2 cache help alleviate the write latency? For instance, the block could first be fully written out to the cache, and the write operation returns.

Now the verification read could be ordered on the back-end, and then the dedup reference created (or the full block in case of a

## Blog Export: c0t0d0s0.org, <http://www.c0t0d0s0.org/>

collision). I don't know if this is reasonable, but just an example of methods to mitigate the impact of additional reads.

Another way to approach this would be to fully write all blocks, and then accomplish dedup asynchronously with a scrub-like operation using block pointer rewrite. This is how I mistakenly imagined dedup working in ZFS originally. Of course this assumes that you have enough extra space.

Finally, there is the small matter of the separately developed Greenbytes deduplication, which I know nothing about. This is just an example of how some contributor may come up with a method more suitable for those worried about hash collisions.

Thanks for this useful post... -cheers, CSB  
Anonymous on Feb 6 2010, 20:51

You wasn't able to use fletcher4 without verify. At the moment you use sha256 in any case, and sha256 is sufficiently strong.  
Anonymous on Feb 6 2010, 21:35

Ahmed, how can you say such thing? Don't you know, what does Qu'ran guide you to? Go read The Book, then do Allah's will!  
Anonymous on Feb 7 2010, 00:50

Most people know ZFS freaking kicks ass. Now that you want to talk about risk,

Risk of ZFS Crypto not making into OpenSolaris-2010.02, and thus L2ARC persistence further delayed to at least half a year from now until the next OpenSolaris release at the earliest.

That's the risk I see.  
Anonymous on Feb 7 2010, 04:39

You have a strange perception of risk. What's the bigger risk: Releasing crypto code too early to make it in a earlier release, or to review the code as most intensive than possible, even when it means, your code gets into a stable release later? Crypto is about trust, and trust can be destroyed within seconds.

By the way: Just look at the authoritative source ... <http://hub.opensolaris.org/bin/view/Project+zfs-crypto/WebHome>  
Anonymous on Feb 7 2010, 09:02

You have a strange perception of risk. What's the bigger risk: Releasing crypto code too early to make it in a earlier release, or to review the code as most intensive than possible, even when it means, your code gets into a stable release later? Crypto is about trust, and trust can be destroyed within seconds.

By the way: Just look at the authoritative source ... <http://hub.opensolaris.org/bin/view/Project+zfs-crypto/WebHome>  
Anonymous on Feb 7 2010, 09:02

From user's perspective, I don't understand why L2ARC persistence needs to depend on ZFS crypto. If you think it is risky to cache the frequent accessed blocks in L2ARC without encryption, the same risk is there for the zpool of hard drives. Brendan Gregg traded off persistence for some additional L2ARC space. He interpreted that L2ARC is an extension of ram, thus missing the only thing flash is superior than ram: persistence.

ZFS Crypto has been delayed longer than any other projects, it is pretty bad, and I don't see it being production ready in OSOL-2010-02.

BTW, what's going on with opensolaris flag days page?  
Anonymous on Feb 7 2010, 14:40

Ahmed posts, "Would you please remove "islamistic" from terror attack... I was very offended."

Joerg posts, "However as i hold the muslim belief in high regard, i changed the part of this article."

I still remember a highly offensive comment on April 10, 2009, explicitly mentioning a radio announcer who was a faithful non-violence advocate and the attribution of violence upon him in conjunction with a crass sexually suggestive comment, presumably because he was a Christian expressing his encouraging feelings with his community regarding their God.

<http://www.c0t0d0s0.org/archives/5458-Trip-to-California-8th-day.html>

"The think i really dislike are the christian radio station... with a moderators saying "... you see how incredible god really is... I'm sure such guys slaps their kids at home and..."

Yes, I have been an avid reader, as well, doing my best to be professional and disregard it... but I still remember it vividly for approximately the past 10 months.

About 10 months ago, I decided to "turn the other cheek", but apparently this was not as effective the path Ahmed and others take.

I'll leave these thoughts up to your intellectual consideration for a reasonably professional and consistent outcome.  
Anonymous on Feb 9 2010, 05:09

thanks for the great article,

It's not about insulting muslims, it's about showing how skewed the risk perception of people is, it's about how idiotic it is to believe, to die due to a terror attack, while not thinking over much more probable risks.

**Blog Export: c0t0d0s0.org, <http://www.c0t0d0s0.org/>**

Anonymous on Jan 10 2012, 08:02