

Forum cracks the vintage passwords of Ken Thompson and other Unix pioneers

Aurich Lawson / Getty

reader comments

66 with 58 posters participating

Share this story

As one of the original versions of Unix, BSD is an ancient operating system. So it shouldn't come as a surprise that it used what are, by today's standards, strange, even ridiculous security. For one, the hashing function protecting passwords, though state of the art 40 years ago, is now trivial to crack. Stranger still, the password hashes of some BSD creators were included in publicly available source code. And then, there are the passwords people chose.

Last week, technologist Leah Neukirchen reported finding a source tree for BSD version 3, circa 1980, and successfully cracking passwords of many of computing's early pioneers. In most of the cases the success was the result of the users choosing easy-to-guess passwords.

BSD co-inventor Dennis Ritchie, for instance, used "dmac" (his middle name was MacAlistair); Stephen R. Bourne, creator of the Bourne shell command line interpreter, chose "bourne"; Eric Schmidt, an early developer of Unix software and now the executive chairman of Google parent company Alphabet, relied on "wendy!!!" (the name of his wife); and Stuart Feldman,

author of Unix automation tool make and the first Fortran compiler, used “axolotl” (the name of a Mexican salamander).

Weakest of all was the password for Unix contributor Brian W. Kernighan: “/././.”—representing a three-character string repeated twice using adjacent keys on a QWERTY keyboard. (None of the passwords included the quotation marks.)

But there were at least five plaintext passwords that remained out of reach. They included those belonging to Turkish computer scientist Özalp Babaoğlu, Unix software developer Howard Katseff, and crucial Unix contributors Tom London and Bob Fabry. But the uncracked hash that seemed to occupy Neukirchen the longest was the password used by Ken Thompson, another Unix co-inventor.

“I never managed to crack ken’s password with the hash Zgh0T0eRm4U9s, and I think I enumerated the whole 8 letter lowercase + special symbols key space,” Neukirchen reported in the above-linked thread, posted to the Unix Heritage Society mailing list. “Any help is welcome.”

From cutting edge to dangerously obsolete

I’ll get to the results later, but first, a discussion of Descrypt, the default hashing algorithm for the BSD 3 operating system. When it debuted in 1979, Descrypt represented the cutting edge of password hashing. Chief among the improvements: it was the first hashing function to use cryptographic salt—which is a randomly chosen text string appended to the password—designed to prevent identical plaintext inputs from having the same hash string. It was also the first to subject plaintext inputs to multiple hashing iterations. With 25 iterations, this so-called key-stretching process significantly increased the time and computation required for attackers to crack the hashes.

Descrypt was deprecated more than 20 years ago, however, as cracking tools grew ever more powerful and better functions came into being. By today's standards, Descrypt is woefully inadequate (though sadly sometimes still used, much to end users' detriment).

Descrypt limits passwords to just eight characters, a constraint that makes it all but impossible for end users to choose truly strong credentials. And the salt Descrypt uses provides just 12 bits of entropy, the equivalent of two printable characters. That tiny salt space makes it likely that large databases will contain thousands of hash strings that attackers can crack simultaneously, since the hash strings use the same salt.

Jeremi M. Gosney, a password security expert and CEO of the password-cracking firm Terahash, told Ars that Descrypt is so weak and antiquated that one of his company's 10-GPU Inmanis appliances (price: almost \$32,000) could besiege a Descrypt hash with 14.5 billion guesses per second (the rigs can be clustered to achieve faster results). The speed of just one rig is enough to brute force the entire Descrypt keyspace—which, due to practical limitations, was about 2^{49} in 1979—in less than 10 hours, and even less time when using cracking tools, such as wordlists, masks, and mangling rules. This site will also crack a Descrypt hashe for as little as \$100.

The weaknesses meant it was inevitable the remaining uncracked hashes Neukirchen posted would be deciphered. But since most of the fellow forum members weren't seasoned password crackers, they seemed to use less efficient techniques. On Wednesday—six days after Neukirchen asked for help—forum member Nigel Williams provided Thompson's plaintext password: "p/q2-q4!" (not including the quotation marks).

Checkmate

It “took 4+ days on an AMD Radeon Vega64 running hashcat at about 930MH/s during that time (those familiar know the hash-rate fluctuates and slows down towards the end),” Williams reported. An AMD Radeon Vega64 is a graphics card and Hashcat is a password-cracking program that takes advantage of the powerful parallel-computing capabilities of graphics cards.

A few hours after Williams’ message, forum member Arthur Krewat provided the passwords for the four remaining uncracked hashes. They were:

- Katseff: graduat;
- Babaoğlu: 12ucdort
- Fabry: 561cml..
- London:...pnn521

Forum members quickly pointed out that Thompson’s password “p/q2-q4!” is the descriptive notation for a common opening move in the game of Chess.

“If I remember right,” another forum member chimed in, “the first half of this password was on a t-shirt commemorating Belle’s first half-move, although its notation may have been different.” Belle was the name of the Chess machine developed by Thompson and Joseph Henry ‘Joe’ Condon. Rob Pike—the forum member who worked on Unix projects while at Bell Labs—then moved on to a different topic not yet brought up in the discussion:

Interesting though it is, though, I find this hacking distasteful. It was distasteful back when, and it still is. The attitudes around hackery have changed; the position nowadays seems to be that the bad guys are doing it so the good guys should be rewarded for doing it first. That’s disingenuous at best, and dangerous at worst.

It's an interesting thought and raises a good point about the ethics of dredging up the passwords of real people. Ultimately, however, I find myself leaning in favor of studying password cracking. Over the past decade, researchers from universities and elsewhere have pored over cracked passwords. The net result: we know much more now than we did a decade ago about choosing a strong password. Forum member Kurt H Maier phrased things well in a response to Pike.

"It's not like we're sitting around rainbow-tabling someone's Macbook," Maier wrote. "This stuff is, at this point, of historical interest. 'How many decades old must a hash be before it's acceptable to decode it' is a valid question worth answering, but comparing this kind of archaeology to active attack is slightly absurd."

Just the same, here's hoping Eric Schmidt and company have changed those vintage passwords.

<https://arstechnica.com/?p=1582657>