Tired of memorizing passwords? A Turing Award winner came up with this algorithmic trick

Passwords are a bane of life on the Internet, but one Turing Award winner has an algorithmic approach that he thinks can make them not only easier to manage but also more secure.

The average user has some 20 passwords today, and in general the easier they are to remember, the less secure they are. When passwords are used across multiple websites, they become even weaker.

Manuel Blum, a professor of computer science at Carnegie Mellon University who won the Turing Award in 1995, has been working on what he calls "human computable" passwords that are not only relatively secure but also don't require us to memorize a different one for each site. Instead, we learn ahead of time an algorithm and a personal, private key, and we use them with the website's name to create and re-create our own unique passwords on the fly for any website at any time.

"I don't even have to remember if I have a password" for a given site, Blum explained last week at the Heidelberg Laureate Forum in Germany. Asked recently by his wife for his password on the REI website, for instance, "I could honestly say, I don't know if I'm registered at REI, but if I am, then my password is...."

Heidelberg Laureate Forum Foundation / C. Flemming

Manuel Blum speaks at the 2015 Heidelberg Laureate Forum.

Essentially, the idea is that your algorithm and key give you an alternate letter or number for each letter in a website's name; that transformed set of values becomes your site-specific password.

Here's an example: One person's key might consist of a six-by-six matrix created from all 26 letters of the alphabet and all 10 digits, but not in the standard order. Rather, the first row consists of the first six letters on a Linotype typesetting machine: E, A, T, O, I and N. The remaining letters are likewise arranged according to the order used in those old devices, followed by the digits 0 through 9.

To transform a website name into a password, the user would use the matrix in combination with an algorithm dictating which letter gets replaced by what. Blum's example used a system following the directions on a compass. Starting with the first letter of the site name, the user goes "north" by one spot on the matrix to find the

replacement for that letter. Next, he finds the second letter of the site name on the matrix and moves "east" to find the value to replace that one, followed by moving "south" to find the replacement for the third, "west" for the fourth, and so on until the entire website name has been encoded.

"Amazon," for instance, becomes "5FHX7E" for a password using this scheme, but you don't have to memorize it -- only the scheme itself.

Blum demonstrated his approach with four audience volunteers, who were able to use it successfully after a few minutes of training.

There are many possible algorithms that could be used -- compass directions are just one example. For sites that require special characters, the user could make it a practice to add a few to the algorithmic results. So he might add "!#$" to the final password each time.

The system would be tough for a hacker to figure out, Blum said.

"As long as you don't give away more than a few passwords, you'll be secure," he said.

The approach clearly requires some upfront work to select and learn your key and algorithm. The point, however, is that you only have to do that once, as opposed to memorizing every password.

For more information, Blum's talk is available online at the Heidelberg Laureate Forum website, and a paper describing the approach was published last fall.