THE NEXT WAVE

Passwordless Authentication & FIDO:
The Future of Security?

by Sibi Chakkaravarthy S, Aswani Kumar Cherukuri, Nandeesh Kumar Kumaravelu, and Aditya Mitra

Password authentication is simultaneously ubiquitous and deeply flawed. Users struggle to remember passwords, companies strive to make them harder to crack, and hackers use a variety of new and old techniques to help them gain access as an "authorized" user and wreak havoc. Rather than trying to improve passwords, this article argues for a change to passwordless authentication, a method that offers a better user experience and more robust security at a lower cost.

Many online platforms, tools, and services use an extremely secure authentication procedure developed by the Fast Identity Online Alliance (FIDO). Specifically, they use FIDO2, which leverages public key cryptography and uses physical keys, lowering the risk of successful phishing attacks to just about zero. Users of FIDO2 include:

- Major platforms (e.g., Google Accounts and Microsoft accounts)
- Social media platforms (e.g., Facebook and Twitter)
- Cloud platforms (e.g., Google Cloud and Oracle Identity Cloud Service)

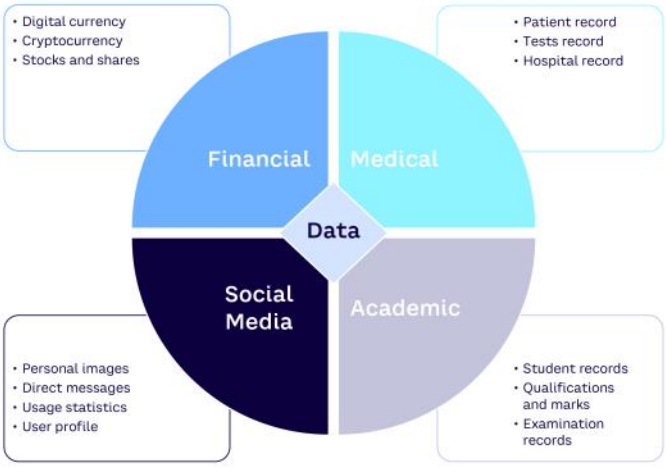


Figure 1. Examples of sensitive data

- Credential management services (e.g., AuthTake and Entrust)
- Cryptographic tools (e.g., Bitwarden, Boxcryptor, and Mickey)
- Gaming services (e.g., Xbox Live)
- Cryptocurrency wallets and trading platforms (e.g., Coinbase, Binance, Authereum, Gemini, DSX, and STEX)

In this article, we provide a brief history of authentication methods, reasons passwords fail to provide strong security in the 21st century, information on FIDO and FIDO2, and an explanation of how passwordless authentication works.

The Evolution of Authentication

There are few places in our lives where data is not a factor — think of our finances, our healthcare, our education, and our social lives (see Figure 1). As the number of attack vectors increase, the attack surface gets broader, and protecting our sensitive data becomes both more important and more difficult.

In the 1960s, researchers at the Massachusetts Institute of Technology (MIT) had access to a time-sharing system created by Fernando Corbató that allowed an IBM mainframe to allocate resources for two tasks at once (it was called the "Compatible Time-Sharing System").[1] To keep individual files being used on the system private (and keep researchers to their allotted four hours per week), Corbató developed a password system. He became known as the father of modern passwords, and passwords became the go-to method for computer security in both personal and corporate spheres.

Over the years, passwords proved a less-than-perfect means of authentication, and other methods evolved.

Personal identification numbers (PINs) were introduced in conjunction with ATM machines in the 1980s. More recently, we have seen biometric and cryptographic authentication methods developed to better secure data. Passwordless authentication with device attestation is another approach (and the focus of this article).

Difficulties with Current Authentication Techniques

With increases in computing power has come an increase in the number and type of attacks on networks and data. Brute-force attacks, password guessing, and phishing have all become common. One problem with passwords is that longer (and therefore more-difficult-to-crack) ones are difficult for people to remember. Another problem is that people are quite willing to use extremely unsafe passwords to make their lives easier. According to SplashData, the top two passwords used from 2011-2018 were "password" and "123456,"[2] and a Wikipedia search for today's most common passwords shows things are not improving.[3]

As a result, many companies have introduced password standards for internal and/or external logins (e.g., must be more than eight characters and must have a number and special character). Some have begun using multi-factor authentication, which requires two or more of the following:[4]

- **Knowledge factors** such as a password or PIN. These are commonly used to authenticate that a username or email address is being provided by the correct person.
- **Possession factors** such as an ATM card or smart card that contain a computer chip that can verify and validate information. These can also take the form of a one-time password sent to the user's mobile phone or a nonsynchronous challenge/response system (the server challenges the user during login with a numerical sequence).
- **Inherence factors** such as a fingerprint, palm print, retina/iris scan, signature recognition, voice recognition, or voice pattern.

Note that the factors in the knowledge category can potentially be extracted via social engineering techniques such as phishing. Additionally, passwords and PINs, once entered, can be extracted by an attacker using technologies such as forward-looking infrared (FLIR). These are essentially thermal cameras that can be used to capture the thermal residue of the user on a keyboard after typing.[5]

Similarly, there are smudges left on the screen of touchscreen devices after entering a password or PIN, to the extent that even a high-resolution camera on a mobile phone can be set to identify the keys involved with authentication. Interestingly, these smudges are difficult to obscure or delete by wiping.[6]

Studies have shown that fingerprint-based authentication systems are vulnerable to a variety of hacks, including artificial clones of fingerprints, printed images, and extracting the fingerprint from the authorized user via social engineering techniques.[7]

Push notification is a type of passwordless authentication and is currently being used by a number of Google and Microsoft notifications. For example, the process of logging into an application from a computer might involve tapping on a notification on the user's smartphone.[8] However, this type of authentication is far less secure than the device attestation used by FIDO (described below). The main reason is that push notifications are usually unencrypted, so they are vulnerable to being read by malicious client-side applications.

FIDO and the Passwordless Alternative

FIDO was founded to reduce the world's overreliance on passwords. It's an open industry association promoting the development and implementation of passwordless authentication and device attestation standards. By promoting and facilitating these approaches, FIDO aims to help companies be less dependent on passwords, help individuals have less password-related stress, and help everyone deal with issues around compromised passwords.

FIDO's passwordless system uses the public key cryptosystem to manage keys. Users can either log in from an authorized device or use a physical security key to log in from another device.

FIDO's Development

The development of the FIDO specification began in 2013 with the launch of the FIDO Alliance.[9] In December 2014, FIDO UAF (Universal Authentication Framework) and U2F (Universal 2nd Factor) were finalized and published simultaneously.

FIDO	FIDO2
Uses UAF and U2F protocols. U2F specifies both a Javascript API and a HID protocol for FIDO. It was never officially recognized as a standard.	Uses WebAuthn and CTAP protocols. WebAuthn specifies a JavaScript API for FIDO2; CTAP specifies a HID protocol to establish communication with external security keys. It was recognized as a standard in December 2018.

Table 1. FIDO vs. FIDO2

In December 2018, after further development, FIDO UAF 1.1 and the Cyber Threat Assessment Program (CTAP) were recognized as international standards by the International Telecommunication Union's Telecommunication Standardization Sector. In March 2019, W3C's Web Authentication API, called "WebAuthn," became an official standard. WebAuthn and CTAP then created the FIDO2 specification. Table 1 lists the shortcomings of FIDO that led to the development of FIDO2.

In October 2021, authentication technology maker Yubico launched YubiKey, which added biometric authentication to FIDO2 security keys, raising the security standard. These keys act as a single, trusted hardware-backed system that allows users to authenticate across multiple devices, operating systems, and applications with the same key.

Also in October 2021, when Google released Chrome 95, it deprecated the use of the U2F API and encouraged websites to use WebAuthn. This essentially marked the end of FIDO in favor of FIDO2 (see Table 2).

The Technology Behind FIDO

FIDO2 comprises two specifications: W3C WebAuthn and FIDO Alliance CTAP. W3C WebAuthn is an API that enables the creation and use of strong, attested, scoped, public key credentials by Web applications for strongly authenticating users.

One or more public key credentials, each scoped to a given WebAuthn Relying Party (RP), are created by (and bound to) authenticators as requested by the Web application. The user agent mediates access to authenticators and their public key credentials to preserve user privacy. Authenticators are responsible for ensuring that no operation is performed without user consent. Authenticators provide cryptographic proof of their properties to RPs via attestation.

The specification also describes the functional model for WebAuthn conformant authenticators, including their signature and attestation functionality. WebAuthn defines the way the RP server communicates with the client device over the Internet. It is essentially a Javascript API for sharing cryptographic keys, challenges, and other required information.[10]

FIDO Alliance CTAP describes an application layer protocol for communications between a roaming authenticator and another client/platform and the binding of this application protocol to a variety of transport protocols using different physical media. The application layer protocol defines requirements for such transport protocols. Each transport binding defines the details of how such transport layer connections should be set up in a manner that meets the requirements of the application layer protocol.[11] CTAP defines the way the Web browser of the client device communicates with the physical security key or the trusted computing platform.

How Does FIDO Work?

FIDO2 specifications are based on device attestation, which ensures that only the device of the authorized user can be used for authentication. There is nothing the user needs to know to authenticate him- or herself, making authentication more seamless and not vulnerable to social engineering. The risk of phishing in this scenario is virtually zero: no unauthorized user can abuse this login method unless he or she physically takes the device and physical security key from an authorized user.

Before authentication, the security keys and/or the device must be registered on the FIDO server against the user's account. The RP communicates with the user's Web browser via W3C WebAuthn, and the browser communicates with the physical security keys via CTAP. Figure 2 shows the components for authentication following FIDO standards.[12]

The Web browser interacts with the RP app server and FIDO server using WebAuthn JavaScript APIs. For

Operating System	Browser	WebAuthn API	CAP			
			USB	NFC	BLE	Platform
Windows	Chrome	Yes	Yes	Yes	Yes	Yes
	Edge	Yes	Yes	Yes	Yes	Yes
	Firefox	Yes	Yes	Yes	Yes	Yes
iOS	Safari	Yes	Yes	Yes	No	Under Dev
Android	Chrome	Yes	Yes	Yes	Yes	Yes
	Edge	Yes	Yes	Yes	Yes	Yes
	Firefox	Yes	Yes	No	Yes	Yes
macOS	Safari	Yes	Yes	No	No	Under Dev
	Chrome	Yes	Yes	No	Yes	Yes
	Edge	Yes	Yes	No	Yes	Yes
	Firefox	Yes	No	No	No	No

*NFC: near-field communication; BLE: Bluetooth Low Energy; Platform: internal authenticator

Table 2. FIDO2 support

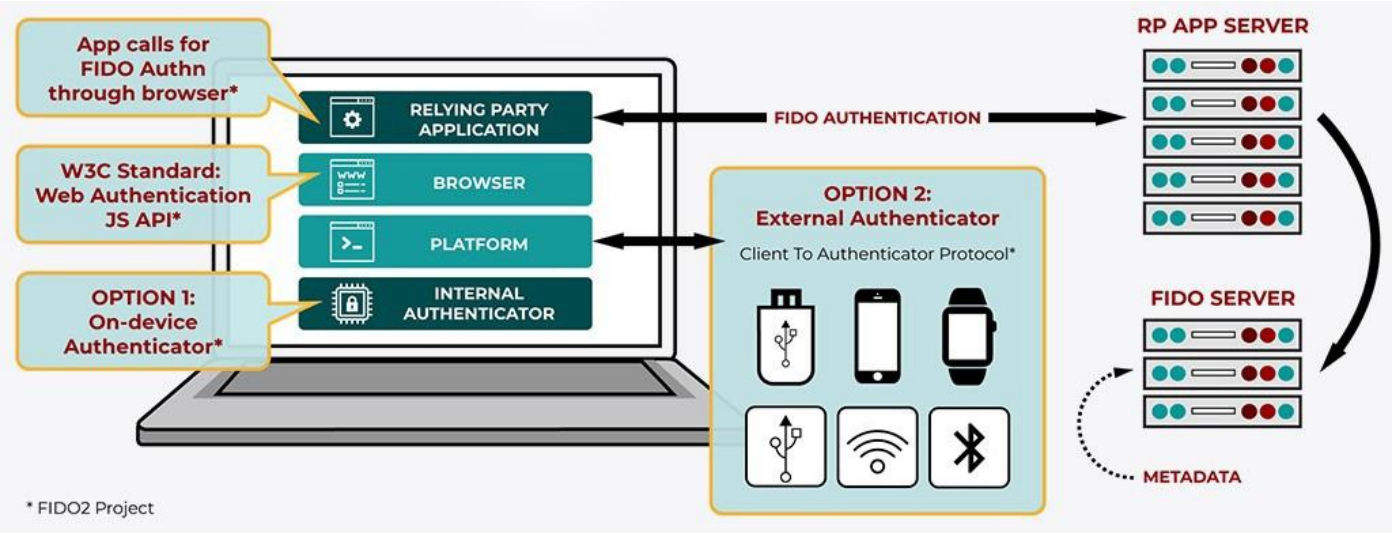


Figure 2. FIDO authentication components (source: FIDO Alliance)

authentication, the program and/or the user has two options: (1) use an on-device (platform) authenticator, which is usually the trusted computing module of the device or (2) use an external authenticator like a physical security key or a smartphone that has been designated as a trusted device. Communication with the authenticator is done using the CTAP protocol. It can work over channels including USB, near-field communication (NFC), and Bluetooth Low Energy (BLE).

When a user asks to add a security key to the account, the RP generates a challenge and sends it along with the RP identifier to the user's browser. The RP identifier must match the server origin as seen by the client. The Web browser checks the certificate and forwards the challenge to the physical security key via CTAP. The key uses the challenge to generate a cryptographic key pair. The challenge is signed with the private key. The signed challenge, the public key, the RP identifier, and a credential ID generated by the physical security key are sent back to the RP. The RP verifies the origin and the signed challenge, stores the public key and the credential ID, and the new the key is registered.

Registering a device or physical security key on a website is simple for a user. He or she navigates to the page where the key is to be added. The website prompts the user to connect the authenticator, the user inserts the key into a USB port, and it is registered.

When the user wants to log in, the RP sends the credential IDs, a challenge, and the RP ID to the browser. The authenticator verifies the RP ID, matches the origin, and matches the credential ID to the security key. The physical security key signs the challenge using the private key. The signed challenge and the RP ID are echoed back to the RP. The RP verifies them and authenticates the user.

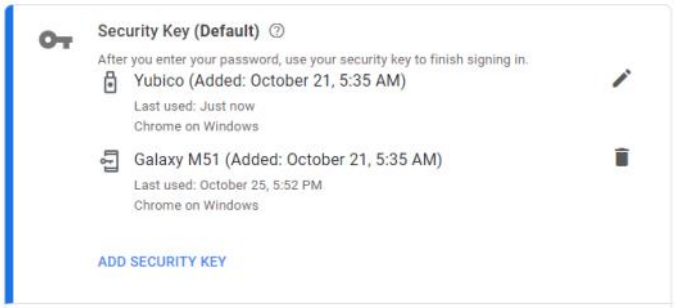


Figure 3. Google Account security page with an added physical security key

The user simply types his or her username into the website login page, is prompted to insert the physical security key into the USB port, and is then automatically authenticated and logged in.

FIDO Applications

It isn't difficult for companies to use FIDO2 authentication. For example, a company that uses Active Directory (AD) could migrate to Azure AD, which supports FIDO2. It could then issue a physical security key in the form of a smart token or ID card to each employee for use in logging in to devices and systems. Companies wishing to make their websites more secure can either set up a FIDO RP server or pay a FIDO-certified provider such as Auth0 or Nok Nok Labs to use one of theirs.

For companies using multifactor authentication, a physical security key can be used as one of the factors. For example, Google uses it as the second step of its two-factor authentication. Figure 3 shows a Google Account security page with added physical security

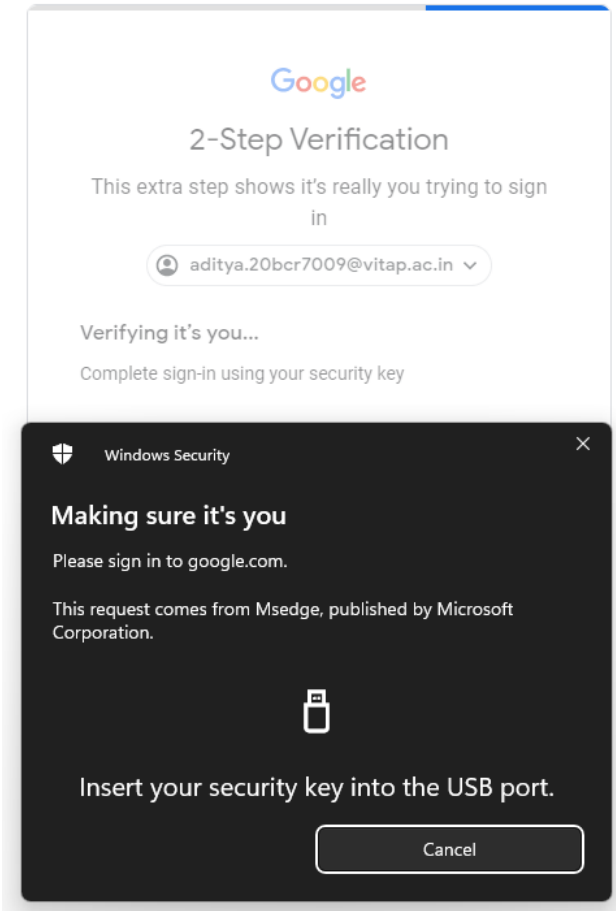


Figure 4. Login page requesting security key

key. When a user wants to log in, he or she must use the security key unless logging in from a trusted device. Figure 4 shows the prompt requesting the security key when a login is attempted.

Looking Ahead

Given the current cybersecurity landscape (especially the number of sophisticated phishing attacks), relying on passwords is not a sound strategy for corporate entities. FIDO2 is significantly more secure, and with most smartphones now shipping with a trusted computing chip in them and Microsoft requiring Trusted Platform Module (TPM) for Windows 11, it's a good bet that FIDO2 is going to be the authentication standard for the foreseeable future.

References

[1]McMillan, Robert. "The World's First Computer Password? It Was Useless Too." *Wired*, 27 January 2012.

[2]"SplashData Reveals 100 Worst Passwords of 2018." *Pinoy Tech Blog*, 17 December 2018.

[3]"List of the Most Common Passwords." Wikipedia, accessed January 2022.

[4]"Authentication Factor" Sumo Logic, accessed January 2022.

[5]Kaczmarek, Tyler, Ercan Ozturk, and Gene Tsudik. "Thermanator: Thermal Residue-Based Post Factum Attacks on Keyboard Data Entry." Cornell University, 26 June 2018.

[6]Aviv, Adam J., et al. "Smudge Attacks on Smartphone Touch Screens." *WOOT'10: Proceedings of the 4th USENIX Conference on Offensive Technologies*. USENIX Association, August 2010.

[7]Mishu, Tanjarul Islam, and Mijanur Rahman. "Vulnerabilities of Fingerprint Authentication Systems and Their Securities." *International Journal of Computer Science and Information Security*, Vol. 16, No. 3, March 2018.

[8]"The Secret Security Wiki: Push Notification Authentication." Secret Double Octopus, 2021.

[9]"History of Fido Alliance." FIDO Alliance, accessed 7 January 2022.

[10]"Web Authentication: An API for Accessing Public Key Credentials Level 2." World Wide Web Consortium (W3C), 8 April 2021.

[11]"Client to Authenticator Protocol (CTAP)." FIDO Alliance, 15 June 2021.

[12]"FIDO2: WebAuthn & CTAP." Fido Alliance, accessed January 2022.

Sibi Chakkaravarthy Sethuraman is Associate Professor in the School of Computer Science and Engineering at Vellore Institute of Technology (VIT), India, where he is also coordinator of the Artificial Intelligence and Robotics Research Center. Dr. Chakkaravarthy Sethuraman is the lead engineer for VISU, an advanced 3D-printed Humanoid Robot developed by VIT. Previously, he was a visiting professor and member of the Artificial Intelligence Lab at Keene State College. Dr. Chakkaravarthy Sethuraman is an active contributor to the open source community and has been published in various security magazines, including Pentestmag *and* eforensics. *He is also a peer reviewer for the Institute of Electrical and Electronics Engineers (IEEE), Springer, the Institution of Engineering and Technology (IET), IGI Global, and Hindawi. Dr. Chakkaravarthy Sethurama is Associate Editor for* International Journal of Cognitive Informatics and Natural Intelligence. *He earned a PhD from Anna University and is a recipient of the DST fellowship. He can be reached at sb.sibi@gmail.com.*

Aswani Kumar Cherukuri is Professor at Vellore Institute of Technology (VIT), India. His research interests include information security and machine learning. Dr. Ch. Aswani Kumar earned the Young Scientist Fellowship from Tamilnadu State Council for Science and Technology and was awarded the Inspiring Teacher Award from The Indian Express *(India's leading English daily newspaper). He has worked on various research projects funded by the Government of India's Department of Science and Technology, Department of Atomic Energy, and the Ministry of Human Resources Development. Dr. Ch. Aswani Kumar has published more than 150 refereed research articles in various national/international journals and conferences and is an editorial board member for several international journals. He is a Senior Member and distinguished speaker of the Association for Computing Machinery (ACM), a member of the Institute of Electrical and Electronics Engineers (IEEE), and Vice-Chair of the IEEE Taskforce on Educational Data Mining. Dr. Ch. Aswani Kumar earned a PhD in informational retrieval, data mining, and soft-computing techniques from VIT. He can be reached at cherukuri@acm.org.*

Nandeesh Kumar Kumaravelu is Associate Professor in the School of Electronics Engineering at Vellore Institute of Technology (VIT), India. His research interests include battery management systems, machine learning, deep learning, and network security automotive electronics. Dr. Kumar Kumaravel earned a PhD from the University of Rome, Italy. He can be reached at kumar.nandeesh@vitap.ac.in.

Aditya Mitra is currently pursuing a bachelor's degree at Vellore Institute of Technology (VIT), India. His areas of interest include physical security and the Internet of Things. He can be reached at adityamitra5102@gmail.com.

Guest editor

Anjali Kaushik

Contributing authors

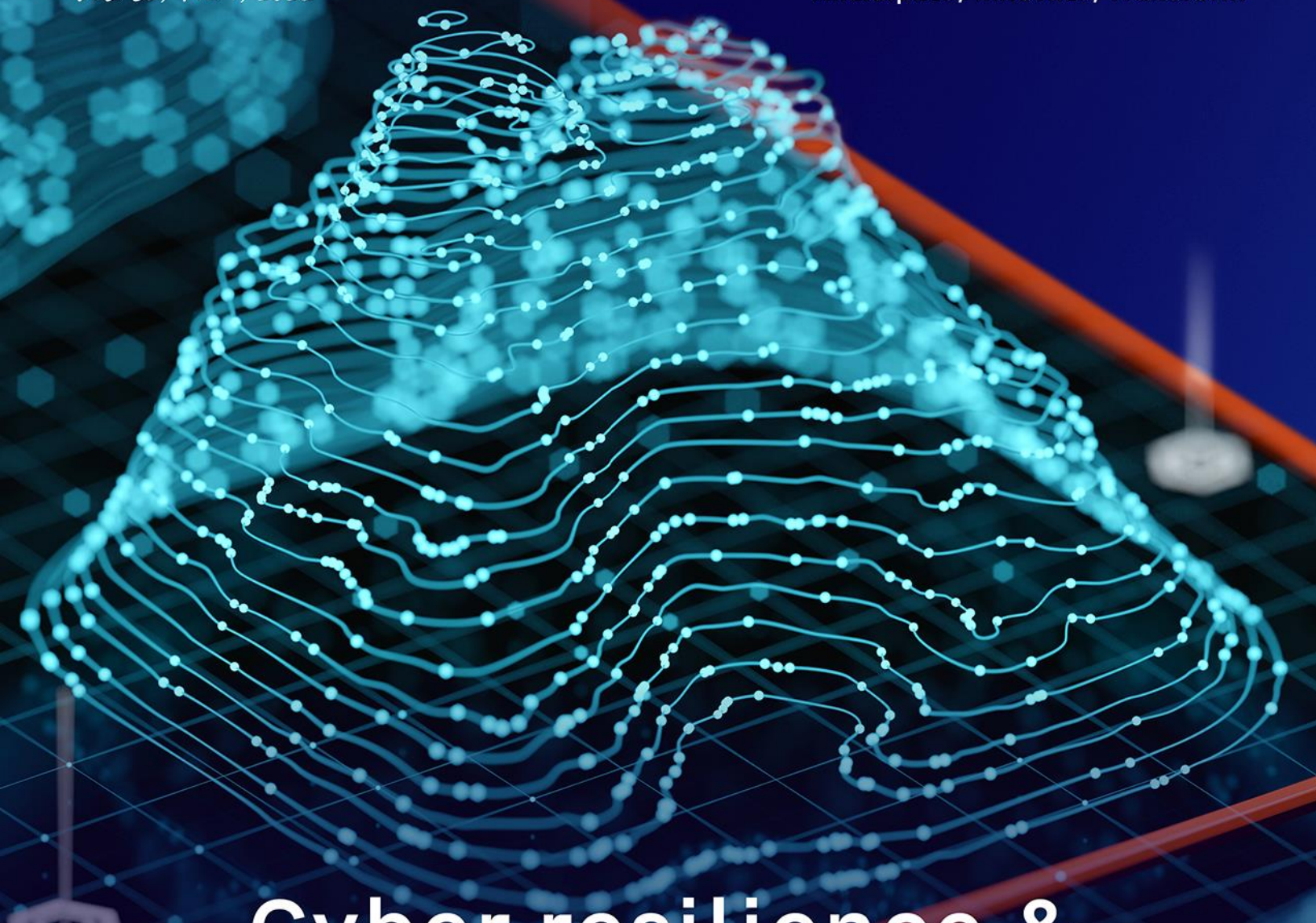
Pradipta Chakraborty
Aswani Kumar Cherukuri
Foivos Christoulakis
Nandeesh Kumar Kumaravelu
Yassine Maleh
Aditya Mitra
Michael Papadopoulos
Richard Phillips
Sibi Chakkaravarthy S
Sivakumar S
David Woodlock

CUTTER
AN ARTHUR D. LITTLE COMMUNITY

AMPLIFY

Vol. 35, No. 1, 2022

Anticipate, Innovate, Transform



Cyber resilience & countermeasures