14. Golbeck, J; Robles, C; Edmondson, M; Turner, K. 'Predicting Personality from Twitter'. 2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing, 149-156. doi:http://10.1109/PASSAT/SocialCom.2011.33. Accessed May 2021. https://ieeexplore.ieee.org/document/6113107.
15. de Montjoye, YA; Radaelli, L; Singh, VK; Pentland, AS. 'Identity and privacy. Unique in the shopping mall: on the reidentifiability of credit card metadata'. Science, 347(6221), 536-539, 2015. doi:10.1126/science.1256297.
16. Hautea, S; Munasinghe, A; Rader, E. 'That's Not Me: Surprising Algorithmic Inferences'. Paper presented at the Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems.
17. Bandyopadhyay, S; Bandyopadhyay, K. 'The European General Data Protection Regulation and Competitiveness of Firms'. Competition Forum, 16(1), 50-55, 2018.
18. Freitas, M; Mira da Silva, M. 'GDPR Compliance in SMEs: There is much to be done'. Journal of Information Systems Engineering & Management, 3(4), 30, 2018.
19. Burkert, H. 'Privacy-enhancing technologies: typology, critique, vision'. In 'Technology and Privacy: the new landscape', MIT Press, 1997, pp.125-142.
20. Pfitzmann, A; Hansen, M. 'A terminology for talking about privacy by data minimisation: Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management'. 10 Aug 2010. Accessed May 2021. http://dud.inf.tu-dresden.de/literatur/Anon_Terminology_v0.34.pdf.

Learning from learning: detecting account takeovers by identifying forgetful users


Sean A McElroy

Sean A McElroy, Lumin Digital

Credential-stuffing attacks are increasing in frequency, allowing threat actors to use data breaches from one source to perpetrate another. While multi-factor authentication remains a crucial preventative measure to protect against credential stuffing, the availability of credential data sets with contact information and the correlation with demographic data can allow threat actors to overcome it through interactive social engineering. Concurrently, alternative defence mechanisms such as network source profiling and device fingerprinting lose effectiveness as privacy-protecting technologies reduce the observable variability between legitimate and fraudulent user sessions.

By measuring a user's increasing familiarity with a web application over time, outliers in use may indicate account takeover fraud. Here, research conducted by the author explores the potential of clickstream data containing logs of users' navigation through a web application as an alternative defence to detecting account takeover activity for digital banking platforms. By identifying when established users are exhibiting learning behaviours in a session, the detection may provide an indicator of compromise.

Human habit

Credential-stuffing attacks take advantage of a common human habit related to passwords: 65% of users reuse the same passwords across multiple systems.[1] Instead of relying on low password complexity, as many brute force attacks do, a credential stuffer reuses usernames and passwords disclosed in previous data breaches against a different target system. Because these attacks only attempt to access a system with a single credential for each user, they often do not trigger account lockout systems that a brute force methodology would. This may be costing US financial institutions alone up to $50m per day.[2]

When sourced from a distributed network, such as a botnet, and with activity spread across several days or weeks, credential-stuffing attacks can be challenging to distinguish from actual users' failed login attempts. Worse, network providers report observing billions of such credential-stuffing attempts monthly and warn that the rate of incidents is increasing substantially.[3]

One effective defence against credential stuffing is implementing a mandatory or risk-based two-factor authentication (2FA) mechanism. Even when attackers might have a username and password pair, if they are unable to receive an out-of-band one-time password or otherwise complete a device or biometric factor of authentication, knowledge alone is insufficient to compromise a user's account. However, threat actors motivated for financial gain (71% of all breaches) are increasingly successful in using stolen credentials, including credential attacks, as a prelude to interactive social engineering techniques against digital banking platforms.[4]

Fraudsters can use the subtle difference in website responses to determine which of their stolen credentials are valid on their target site, forge a Caller ID name to mimic the targeted financial institution, and pose as a call centre representative to induce a victim to read back a one-time password so it can be typed in by the attacker. Such methods are often successful because many such systems' defences hinge on a strong perimeter. However, once authenticated, a threat actor may have access to steal identity information or move money.

Fraudsters have been successful by circumventing perimeter defences completely by targeting the weakest link – the human element. While 'Nigerian prince' scams may have limited effectiveness in 2020, by posing as known acquaintances on social media or the institution itself, fraudsters remain capable of socially engineering credentials from users.

Key targets

Banks and credit unions have long been targets of attackers, ranging from individuals to highly sophisticated and well-funded nation-state actors.[5,6] In response to these threats, the financial services industry follows a comprehensive overlay of regulatory requirements and regular examinations from government agencies. A rich ecosystem of commercial solution providers offers integrable services for digital banking platforms. These products include perimeter defences, such as next-generation firewalls and stronger factors of authentication. However, such methods often presume the victim is not a party to the attack – which is the case when they are compromised through interactive social engineering.

Recently, the idea of 'continuous authentication' has gained the interest of security practitioners. Continuous authentication considers a broad set of a user's behaviours profiled over time to provide additional data points that score the certainty that the user's identity matches expected activities. This idea is appealing because it incorporates or relies on a rich context beyond individual knowledge or possession-based factors of authentication.

"Continuous authentication considers a broad set of a user's behaviours profiled over time to provide additional data points that score the certainty that the user's identity matches expected activities"

As digital banking continues to shift to mobile form factors, the device sensors on smartphones allow for intriguing use cases, from motion-based profiling via accelerometer readings to gait analysis, as users move about the physical world.[7] Profiling users in this manner can be problematic, since device manufacturers continue to make changes to limit continued access to these sensors to preserve privacy and conserve battery power.

General solutions that operate at the physical device layer may not be consistently accessible. However, the principle of establishing a continuous authentication methodology within an application may have merit in identifying threat actors attempting to take over accounts through credential-stuffing attacks.

Patterns of behaviour

When reviewing audit logs of a consumer's account takeover, for those incidents where an attacker interactively accessed an account with a stuffed credential, certain patterns of behaviour are anecdotally apparent. Users who are familiar with a system appear to exhibit goal-directed behaviour when navigating web interfaces. In contrast, users who access a digital banking platform for the first time appear to meander through screens, discovering features and paths before requesting an action.

Software usability is a concept familiar to industrial engineering and is formally defined by ISO/IEC 25010:2011. A component of usability is learnability, which is defined in part as the "degree to which a product or system can be used … to achieve specified goals of learning to use the product or system with effectiveness".[8]

Learnability is certainty important for a product. Design teams should measure the time it takes a user to become familiar with and quickly accomplish tasks using the technology. A rich history of physiological response measurement and analysis exists in industrial engineering and human factors research.[9] Measuring learnability has also become important in software engineering. Advances in eye-tracking, expression monitoring and machine learning have extended these techniques that companies incorporate into their products' lifecycles.[10] As an internationally and well-defined usability concept, with mature assessment techniques established, measuring learnability may provide an opportunity not only to measure the time it takes for a user to learn how to use a system's features, but it may also indicate 're-learning'.

Prior academic research has investigated measures for indications of re-learning to identify potential 'learning issues' that may stem from system usability design flaws.[11] Machine learning and anomaly detection have long held the interest of researchers and practitioners alike in detecting digital banking fraud, with varying levels of success. However, an opportunity exists to determine whether the identification of learning behaviours among frequent users could be indicators, not of appli-

cation usability flaws, but fraudulent activity. If applied in the context of online financial services, it could provide a way to overtly deny account takeover attempts or covertly classify suspect sessions for fraud monitoring.

Predicting fraud

To examine the question of whether the detection of learning behaviours in user activity data can predict fraud, a digital banking solution provider provided access to a repository of anonymised 'clickstream' data. The data was labelled based on whether a session was associated with a financial institution report of fraudulent activity. A full copy of this clickstream data was obtained in CSV format that contained an incrementing database identifier for each user session, the click date and time, a GUID representing the user, the URL path for the HTTPS request, the user's IP address, the user-agent string and a GUID representing the unique session for the user.

Importantly, the platform's application server generated these logs – not third-party analytics services that operate solely on the user's client, such as Google Analytics. Because many user-agents and browser extensions block or degrade the effectiveness of user profiling and clickstream tracking for enhanced privacy, using those sources may skew the analysis of specific user segments.

A variety of constituents utilise digital banking, including end-user consumers and account aggregators that log in on behalf of users using their credentials. Also, synthetic transaction monitoring tools, which authenticate with test accounts to measure the platform's performance and responses, can skew the data set. Similarly, other sources of automated access, such as dynamic application security testing tools, may represent a significant number of sessions over time and may not be representative of user behaviour. For this reason, the IP address and user-agent strings were used to filter out aggregation and performance-monitoring platforms. Similarly, tools were used, including sed, grep, cut and grepcidr, to exclude activity from the institution and platform provider itself to exclude testing and support activity from this analysis.

A series of scripts was written to normalise paths for path analysis, including replacing GUID and numerical resource identifiers to placeholders, such as condensing /accounts/activity/f5ca9a9c-b806-44e3-bfa2-fc791c4868cb to /accounts/activity/*GUID*. Because the IP address and user-agent data are unnecessary for this experiment, it was securely deleted once IP-based filters were applied.

By grouping 64,747,197 individual navigation events in 19.4GB of logs, such as 'User navigated to /accounts', into 10,387,421 unique session identifiers, session-indexed paths were prepared for the experiment. The resulting population of session events was split into two data sets. The first set contains sessions of users who have logged in at least 10 times prior, but four times in the preceding 90 days (the 'infrequent users' sessions' set). The second set contains sessions for users who logged in at least 10 times prior and four or more times in the preceding 90 days. By dividing the population of user sessions by the anticipated learning behaviour activity of each segment's population, the relative strength of the detection method for each could be measured.

Learning behaviour

To construct a potential measure of learning behaviour, an algorithm was created, based on observation and analysis of the first-time sign-on data and several observations were made when searching for potential differences between infrequent user sessions and engaged users' sessions. First, the average number of pages visited per session was higher (9.3 vs 6.3) among infrequent users. Second, infrequent users were more likely to navigate areas providing infrequently changed settings, such as user profile information (7.0% vs 3.2% sessions). Third, engaged users were more likely to modify banking alerts and log in to view or change alerting settings (3.9% vs 1.6% of sessions).

Unexpectedly, in the search for 'meandering' behaviours, where a user returned to the same pages more than once in a given session, the average number of total pages versus unique pages visited per session was consistent at a 2.0 ratio among all user classes. A significant number of users accessing the system (52.9%) log in and view information on a comprehensive dashboard and log out after reviewing only one or two total pages per session. For this reason, such sessions lacked the resolution required to discern whether users were familiar with a system or did not need to learn it because their reason to log in was satisfied without needing to navigate through additional screens.

Based on the observations mentioned above, an algorithm was designed and implemented to produce a 0 or 1 as to whether it detected a learning behaviour within each of the infrequent user and frequent user data sets when a session met three of the four following criteria:

1. The total session time divided by the number of pages was higher than an average of five seconds per page.
2. The total pages visited in the session were higher than seven.
3. Whether users visited help content or entered non-transactional search terms in the application search textbox at the top of the digital banking application.
4. Whether users viewed profile pages or user settings pages that contained options that are infrequently changed.

Known indicators

Known potential indicators of fraud, such as adding new external accounts or bill pay payees, modifying contact settings, or disabling alerts, were not included in the algorithm. These indicators were excluded, since the purpose of the experiment was to determine if the

Duration in seconds	Cum. prior session count	Last 90 day prior session count	Cum. prior session seconds	Profile visited	Alerts visited	FAQ visited	Edge depth	Avg. duration per page	Data set	Learning behaviour	Fraud
5083	1	1	5083	1	1	1	236	21.53	0	YES	NO
33	2	2	5116	0	0	0	6	5.5	1	NO	NO
517	1	1	517	0	0	0	11	47	0	NO	NO
41	2	2	558	0	0	0	3	13.66	1	NO	NO
683	3	3	5799	0	1	0	29	23.55	1	YES	NO
39	1	1	39	0	0	0	3	13	0	NO	NO
23	2	2	62	0	0	0	4	5.75	1	NO	NO
107	1	1	107	0	0	0	11	9.72	0	NO	NO
139	3	3	201	0	0	0	7	19.85	1	NO	NO
252	3	3	810	1	1	0	9	28	1	YES	NO

Table 1: Sample of prepared session data set file.

detection of general learning behaviour in established and engaged users serves as a fraud detection mechanism, not to detect fraud-related activity directly through this experiment. This algorithm was implemented as a custom .NET Core program using C# to process the source data set into various reports for analysis in Microsoft Excel.

The final data set was formatted in a single result file with each record representing a single session. Each record contained fields for the dimensions required for this analysis, including the user's navigational path during the session, as exemplified in Figure 1.

Before the algorithm was applied to the labelled data set of fraudulent sessions (n=62), an analysis was made of each dimension to visually determine whether each supports the hypothesis that each may indicate a learning behaviour exhibited by users with less familiarity. The time spent per page between user navigation events aligned with expectations that first-time users would spend more time navigating through the site than experienced users frequently logging in, presumably because new users are reading more text and are learning the visual cues of the system. While this behaviour differentiated around the 8sec average for the tested web application, 41% of sessions had an average of 7secs per page or less, as represented in Figure 2. This suggests that for user sessions where the user is actively engaged and average navigation time per page is low, this dimension may be significant to separating those who are familiar with the application from those who are still learning how to use it.

Similarly, evaluating the number of pages a user of a given data set navigated to in a given session, visualised as a cumulative percentage of sessions, suggested the second criterion of the detection algorithm was useful. As expected, first-time users explored more of the digital banking web application, with the last 10% of first-time logins viewing 20 or more pages but 90% of returning frequent users viewing eight pages or fewer. Because different functions, such as editing a scheduled banking transfer, require a minimum number of pages regardless of user familiarity, the graph of this dimension suggests that its value as an independent assessment criterion is limited to goal-directed sessions where the user's intent and action can be discerned and measured categorically. For the purposes of this research, however, these page views are averaged by the data set as depicted in Figure 3.

For the third criterion, the measure of the percentage of user sessions for each data set of first-time, infrequent and frequent frequent users also appeared to fit the hypothesis. First-time users were 5.5 times more likely, and infrequent users were 2.3 times more likely, to view FAQs or to search for content in an application search textbox than frequent users (2.2% vs. 1.0% vs. 0.3% of sessions, respectively). This visualisation supports the supposition that users who use the system more frequently have learned how to use features that new or infrequent users may explicitly need help with.

Finally, first-time users were five times more likely to navigate to rarely used pro-

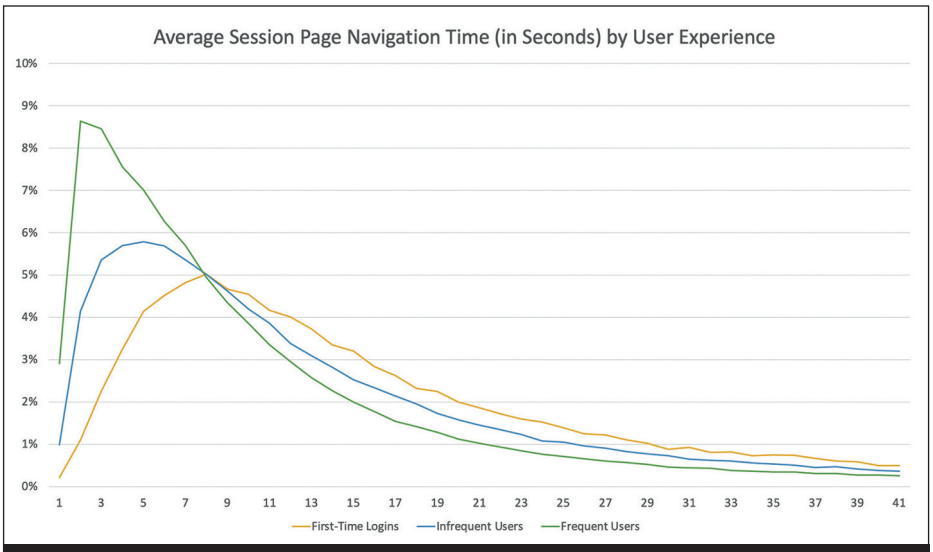

Figure 1: Learnability as measured by page navigation time.

file page settings. Infrequent users were twice as likely to do the same, as compared to frequent users of the digital banking platform (2.2% vs 1.1% vs 0.35%, respectively). Logically, users verify their information and explore these areas, but over time, rarely revisit them unless they update settings that affect the overall behaviour of their experience, rather than research specific transactions. Because of the stark 5x difference in the activity, and given that fraudulent sessions often view or change contact settings under these profile areas, this measure might be an especially strong candidate to differentiate between legitimate and fraudulent sessions for frequent users of digital banking.

Odds ratios

Both source data sets were processed to determine two odds ratios (OR) with the algorithm prepared. The odds ratio is the ratio of the probability of an event occurring in one group to the odds of it occurring in another.[12] To test the hypothesis that learning behaviours, as detected by the algorithm, may be indicators of fraudulent behaviour, the odds ratio for the population of a data set must be greater than 1.0. The values used to calculate the odds ratio to test this hypothesis can be visualised as a 2x2 table (Table 2).

By applying the learning behaviour algorithm to the population of sessions in a data set and calculating these four independent values, the odds ratio would be $OR=(F_L/G_L)/(F_N/G_N)$.

In this context, the odds ratio is the ratio of the probability that the algorithm will detect a learning behaviour in a fraudulent session to the probability that it will not detect a learning behaviour in a session later reported as fraudulent. For example, if 3% of sessions flagged as containing a learning behaviour were fraudulent, but only 2% of sessions of those not flagged were fraudulent, the OR1 would be 1.5. If this were true, the odds ratio would be greater than 1.0, which would indicate that learning behaviours are associated with an increased incidence of fraud. If the OR were <1.0, the algorithm would not be an effective method of detecting the learning behaviours hypothesised to be exhibited by users with less familiarity or usage of the system. The findings must show OR > 1.0 to suggest this mechanism may be useful for detecting learning behaviours that are also indicative of digital banking fraud.

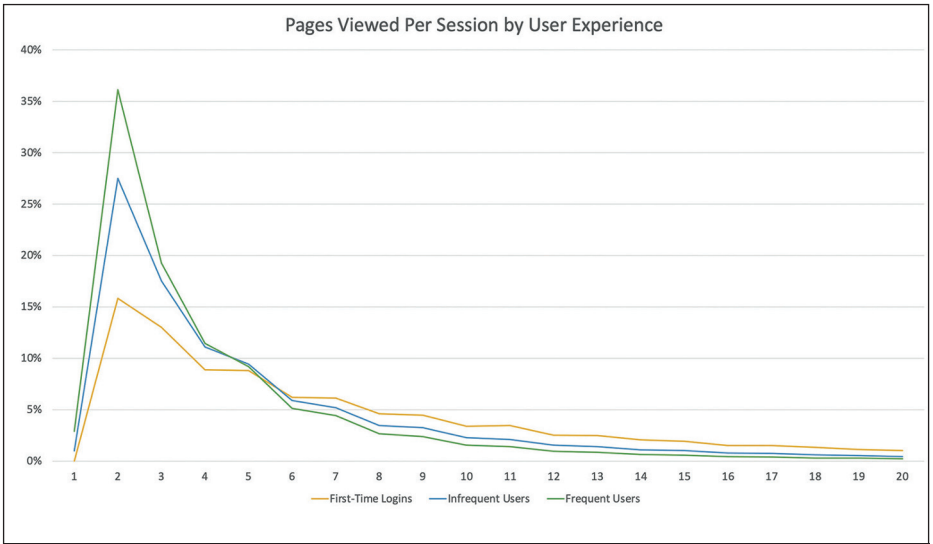

Figure 2: Learnability as measured by per-session page depth.

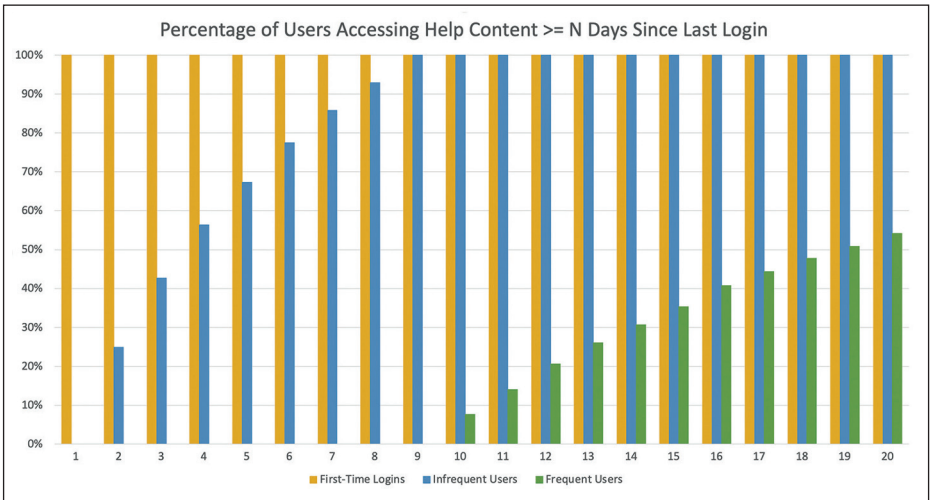

Figure 3: Variances in help content use by user class.

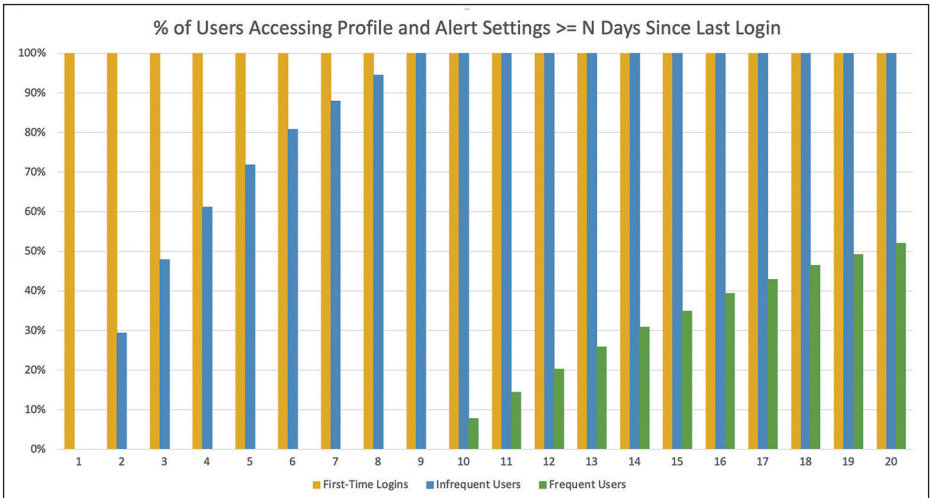

Figure 4: Similar variances in profile and alert settings by user class.

	Fraud Reported	No Fraud Reported
Algorithm detected learning behaviour	FL	GL
No learning behaviour detected by the algorithm	FN	GN

Table 2: Odds ratio parameters derived from the experiment design.

After the experiment, the learning behaviour detection algorithm's output resulted in the coefficients shown in Table 3.
As a result, the odds ratio for infrequent users' sessions was OR1 = 29.92. For frequent users' sessions, the ratio was OR2 = 205.38. Both odds ratios are greater than 1.0, indicating the detection of learning behaviours using the dimensions defined in this research can indicate fraud. Moreover, since OR2 > OR1, this strongly suggests that the ability to detect fraud is improved for users who frequently utilise the system.

User engagement

Increasing user engagement with the application has many commercial benefits for the provider in retaining customers and expanding relationships, as the institution can better know and cater to the user's needs. Moreover, this research indicates there is a mutual benefit in that – as users learn a system and regularly engage with it, they improve their knowledge of how to use it. Consequently, they provide the platform data that can be used to indicate account takeover fraud risk.

As this research demonstrates, application usage data can be leveraged to detect fraud by identifying learning behaviours. Collecting the requisite data, operationalising the analysis of it and properly leveraging results in post-detection actions is critical to realising the value of this detection methodology.

This research used data widely available in web page analytics solutions – namely user and session identifiers, date and timestamps, and page navigation paths. Many applications log this data in server-side repositories for diagnostic purposes. However, to leverage auditing to mitigate fraud losses, this user session data must be analysed after the session has ended but within a period sufficient to allow for analysis of flagged user sessions and to stop a loss before it is realised. An appropriate place to add learning behaviour detection routines may be an enrichment step in an existing logging pipeline, such as through a custom Logstash filter.

Not all web applications will benefit from this algorithmic detection method. Applications that lack feature breadth or those that provide only linear flows, such as a checkout process where there is little optionality or which provide little user choice will not require users to learn how to navigate through or use the system to a degree sufficient enough to detect distinct learning behaviours. Application security teams interested in identifying learning behaviour dimensions and designing quantitative measures should collaborate with product design teams on initial and ongoing efforts. Measuring usability and learnability are concerns for product interface designers, and the feedback from focus groups and quantitative usability measures can inform security implementations of this technique.

While this research focused on learning behaviours broadly, they likely vary significantly by user segment and medium. For instance, an elderly consumer accessing a web application on a desktop computer in an office may exhibit different behaviours than an active teenage user on a mobile device. Accounting for the variances in application usage, these differences in form factors, timing and the inherent time it takes different groups to learn and become familiar with an application may produce a more accurate model for detecting the differing behaviours a subsequent account takeover session would exhibit.

		Fraud reported	No fraud reported
Infrequent users' sessions	Algorithm detected learning behaviour	10	50,716
	No learning behaviour detected by the algorithm	10	1,517,423
Frequent users' sessions	Algorithm detected learning behaviour	31	32,178
	No learning behaviour detected by the algorithm	16	3,304,295

Table 3: Odds Ratio parameters calculated from experiment results

Measuring tools

Usability and learnability measuring tools in the context of user interface design and product management may supplement security-driven modelling of learning behaviour. For instance, at least one such commercial tool provides for the detection of frustrated users through "rage, dead and error clicks or high rates of abandoned forms".[13] These behaviours, which may not be recorded by server-side application event auditing mechanisms, if incorporated, may further enhance the modelling of learning behaviour for the purposes of fraud detection.

Account takeover fraud will be a significant challenge for online service providers for a long time to come. With so many avenues available to threat actors to circumvent technical controls by socially engineering providers and their users, a layered approach to continuously authenticating and assessing users is necessary to recognise and stop the fraud losses that account takeovers can create. This research demonstrates that by measuring when a user is learning the system, that measurement can help detect fraud when such behaviours are observed in sessions for users who have established familiarity or mastery of a user interface.

Notably, the results of this fraud detection method in the context of digi-

tal banking can be implemented with existing application usage data common to many types of web applications. Formulating measures of learning behaviour is not so complicated as to require data analytics expertise. With consistent and comprehensive server-side application logging, basic analytical tools can implement this technique. Operationalising this detection method at scale generally does not require esoteric or expensive machine learning or anomaly detection models or tools. Security analysts have an opportunity to work closely with product managers, user interface designers and engineering teams to mitigate the potential for fraud by collectively learning how to recognise when users are learning.

About the author

As co-founder of Alkami Technology, Sean McElroy helped protect millions of consumers across online banking platforms. Currently, as the CISO of Lumin Digital (https://lumindigital.com), he is responsible for cyber security, risk and compliance programmes to scale the next generation of cloud-based digital banking. McElroy earned a Master of Science in Information Security Engineering from the SANS Technology Institute, has BBAs in Management and MIS from the University of Oklahoma, and holds multiple security certifications.

Resource

Source code for data transformation is made available under the GPL 3.0 licence at https://github.com/seanmcelroy/clickstream-fraud.

References

1. 'IBM Cyber security and Privacy Research'. The Harris Poll/IBM, Dec 2018. Accessed May 2021. https://newsroom.ibm.com/Cyber security-and-Privacy-Research.
2. 'Credential Spill Report'. Shape Security, 2018. Accessed May 2021. https://info.shapesecurity.com/credentialspillreportcyberwire.html.
3. Townsend, K. 'Credential stuffing attacks are reaching DDoS proportions'. SecurityWeek, 24 Sep 2018. Accessed May 2021. www.securityweek.com/credential-stuffing-attacks-are-reaching-ddos-proportions.
4. 'Data Breach Investigations Report'. Verizon, 2019. Accessed May 2021. https://enterprise.verizon.com/resources/reports/dbir/2019/introduction/.
5. Sheridan, K. 'Capital One: what we should learn this time'. Dark Reading, 2 Aug 2019. Accessed May 2021. www.darkreading.com/cloud/capital-one-what-we-should-learn-this-time/d/d-id/1335426.
6. 'APT 38: Un-usual Suspects'. FireEye, 2018. Accessed May 2021. www.fireeye.com/content/dam/fireeye-www/current-threats/pdfs/pf/apt/rpt-apt38-2018.pdf.
7. Abuhamad, M; Abusnaina, A; Nyang, D; Mohaisen, D. 'Sensor-based continuous authentication of smartphones' users using behavioral biometrics: a contemporary survey'. IEEE Internet of Things Journal, 28 Aug 2020. Accessed May 2021. https://ieeexplore.ieee.org/abstract/document/9179700.
8. 'Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models'. ISO/IEC, 1 Mar 2011. Accessed May 2021. www.iso.org/obp/ui/?_escaped_fragment_=iso:std:iso-iec:25010:ed-1:v1:en.
9. Akamatsu, M; Green, P; Bengler, K. 'Automotive technology and human factors research: past, present, and future'. International Journal of Vehicular Technology, 27, 4 Sep 2013.
10. Juin, B; Diah, N. M; Ismail, M; Adam, NL. 'Eye tracking parameters for measuring learnability in mobile-game-based learning'. 2017 IEEE Conference on e-Learning, e-Management and e-Services (IC3e). pp.49-54. Miri: IEEE.
11. Marrella, A; Catarci, T. 'Measuring the learnability of interactive systems using a petri net based approach'. Proceedings of the 2018 Designing Interactive Systems Conference, pp.1309-1319. Hong Kong: Association for Computing Machinery.
12. Buis, ML. 'Logistic regression: when can we do what we think we can do?' 29 May 2017. Accessed May 2021. https://maartenbuis.nl/presentations/index.html.
13. 'Frustrated Sessions'. FullStory. Accessed 23 Aug 2020. https://help.fullstory.com/hc/en-us/articles/360020828013-Frustrated-Sessions.

Deconstructing the SolarWinds breach

Lavi Lazarovitz, CyberArk Labs

Lavi Lazarovitz

In 2021 we've already seen a number of far-reaching cyber attacks. The now infamous SolarWinds breach is still causing ramifications for the businesses involved, and the recent hack on Microsoft exchange servers has reportedly put more than 3,000 UK email servers at risk.[1]

And they just keep coming. It won't be long until we see another attack of