Paweł Kuraś
email: pkuras@wsb-nlu.edu.pl
ORCID: 0000-0002-8658-0821
Wyższa Szkoła Biznesu – National Louis University

Anna Turek
email: aturek@prz.edu.pl
ORCID: brak
Politechnika Rzeszowska im. Ignacego Łukasiewicza

Maciej Gacek
email: gacekmaciej99@gmail.com
ORCID: brak
Wyższa Szkoła Informatyki i Zarządzania w Rzeszowie

Aplikacja zasad Design Thinking w kryptografii

Application of Design Thinking Principles in Cryptography

DOI: 10.66935/978-83-65926-17-3.21

Cytuj jako: Kuraś, P., Turek, A., Gacek, M. (2025). Aplikacja zasad Design Thinking w kryptografii. W: J. Lizak, M. W. Sidor (red.), *Nowe wyzwania w naukach społecznych i technicznych: podejście interdyscyplinarne: T. 2* (s. 1-). Wydawnictwo WSB-NLU.

Streszczenie: Współczesne systemy kryptograficzne, mimo wysokiego poziomu zaawansowania matematycznego, często zawodzą na płaszczyźnie interakcji z użytkownikiem końcowym (Human-Computer Interaction). Celem niniejszego artykułu jest analiza możliwości zastosowania metodologii Design Thinking w procesie projektowania bezpiecznych narzędzi cyfrowych. W pracy postawiono tezę, że uwzględnienie perspektywy użytkownika na wczesnym etapie projektowania (*security by design*) przy użyciu narzędzi empatyzacji i prototypowania, znacząco podnosi realny poziom bezpieczeństwa

systemów IT. W części badawczej przeprowadzono eksperyment z udziałem grupy testowej (N=30), mający na celu zaprojektowanie interfejsu zarządzania kluczami kryptograficznymi zgodnie z pięcioetapowym procesem Design Thinking. Wyniki wskazują, że podejście zorientowane na człowieka redukuje liczbę błędów krytycznych popełnianych przez użytkowników oraz zwiększa zrozumienie mechanizmów zabezpieczeń. Artykuł omawia również bariery we wdrażaniu interdyscyplinarnego podejścia łączącego inżynierię bezpieczeństwa z projektowaniem User Experience (UX).

Abstract: Modern cryptographic systems, despite their high level of mathematical sophistication, often fail in terms of Human-Computer Interaction (HCI). The aim of this paper is to analyze the applicability of Design Thinking methodology in the process of designing secure digital tools. The paper posits that incorporating the user's perspective at an early design stage (*security by design*) using empathizing and prototyping tools significantly increases the real security level of IT systems. The research section describes an experiment involving a test group aimed at designing a cryptographic key management interface following the five-stage Design Thinking process. The results indicate that a human-centric approach reduces the number of critical errors made by users and enhances the understanding of security mechanisms. The article also discusses barriers to implementing an interdisciplinary approach combining security engineering with User Experience (UX) design.

Słowa kluczowe: Design Thinking, kryptografia, użyteczność, User Experience, bezpieczeństwo informacji.

Keywords: Design Thinking, cryptography, usability, User Experience, information security.

Wprowadzenie

W dobie powszechnej cyfryzacji kryptografia przestała być hermetyczną domeną agencji wywiadowczych i matematyków akademickich, stając się fundamentalnym spoiwem współczesnej infrastruktury informatycznej. Zgodnie z *Prawem Kerckhoffsa*, które sformułował Auguste Kerckhoffs (1883), bezpieczeństwo systemu nie powinno opierać się na tajności algorytmu, lecz na tajności klucza. Rozwój ten jest bezpośrednią kontynuacją rewolucji zapoczątkowanej przez Diffiego i Hellmana (1976), którzy wprowadzili koncepcję kryptografii klucza publicznego. Współczesne protokoły, od TLS 1.3 zabezpieczającego bankowość elektroniczną, po Signal Protocol opisany przez Cohn-Gordona i in. (2017) w komunikatorach, realizują ten postulat przy użyciu zaawansowanej matematyki dyskretnej.

Niemniej jednak, paradygmat czysto techniczny napotyka na barierę w punkcie styku z człowiekiem. Jak celnie zauważył Bruce Schneier (2004), „amatorzy atakują systemy, profesjonaliści atakują ludzi". Nawet algorytm o teoretycznej odporności na ataki rzędu $O(2^{256})$, taki jak AES-256, staje się bezużyteczny w obliczu błędów poznawczych użytkownika. Zjawisko to można opisać formalnie, definiując efektywny poziom bezpieczeństwa systemu S_{eff} jako funkcję siły kryptograficznej $C_{strength}$ i prawdopodobieństwa błędu ludzkiego $P(E_{human})$:

$$S_{eff} = C_{strength} \times \left(1 - P(E_{human})\right)$$

Gdy $P(E_{human}) \to 1$ (np. zapisanie hasła na kartce, ignorowanie ostrzeżeń SSL), $S_{eff} \to 0$, niezależnie od wartości $C_{strength}$.

Niniejszy artykuł proponuje zmianę paradygmatu poprzez rygorystyczną aplikację zasad *Design Thinking* (myślenia projektowego) w inżynierii kryptograficznej. Celem jest zminimalizowanie luki poznawczej między modelem matematycznym a modelem mentalnym użytkownika, co w literaturze zdefiniowali Whitten i Tygar (1999) mianem *Usable Security*.

Kryptograficzny rygor a ograniczenia poznawcze

Fundamentem bezpieczeństwa cyfrowego jest teoria informacji opracowana przez Claude'a Shannona (1948). Shannon zdefiniował entropię H jako miarę niepewności (informacji). W kontekście haseł, entropia H dla hasła o długości L wybranego ze zbioru znaków o mocy N wyraża się wzorem:

$$H = L \cdot \log_2(N)$$

Dla $N = 94$ (drukowalne znaki ASCII) i $L = 8$, $H \approx 52.4$ bity.

Z perspektywy kryptoanalizy dążymy do maksymalizacji H, na co zwracają uwagę Bonneau i in. (2012). Jednakże badania psychologii poznawczej, w tym słynna praca Millera (1956), wskazują na drastyczne ograniczenia pamięci roboczej człowieka („magiczna liczba siedem plus/minus dwa"). Wymuszanie wysokiej entropii (np. losowe ciągi znaków) prowadzi do konfliktu z prawami psychologicznymi (Prawo Yerkesa-Dodsona), zmuszając użytkownika do stosowania niebezpiecznych heurystyk opisanych przez Daniela Kahnemana (2012).

W systemach asymetrycznych, takich jak RSA stworzony przez Rivesta, Shamira i Adlemana (1978), problem staje się jeszcze bardziej złożony. Bezpieczeństwo opiera się na

trudności faktoryzacji dużych liczb, gdzie klucz publiczny (n, e) i prywatny (n, d) spełniają relację kongruencji:

$$\begin{aligned} m^{ed} &\equiv m \pmod{n} \\ \text{gdzie: } ed &\equiv 1 \pmod{\phi(n)} \end{aligned}$$

oraz $\phi(n)$ to funkcja totient Eulera.

Dla przeciętnego użytkownika operacje w ciele $\mathbb{Z}_n$ są całkowicie abstrakcyjne, a brak wizualnej metafory dla tych działań prowadzi do błędów w zarządzaniu kluczami, co wykazali Garfinkel i Miller (2005) w swoich badaniach nad S/MIME.

Modelowanie luki użyteczności (The Usability Gap)

Aby zrozumieć naturę problemu, musimy odwołać się do modelu „Gulf of Execution and Evaluation" autorstwa Normana (1988). W kontekście kryptografii, luka ta jest przestrzenią między stanem matematycznym systemu a jego percepcją przez użytkownika.

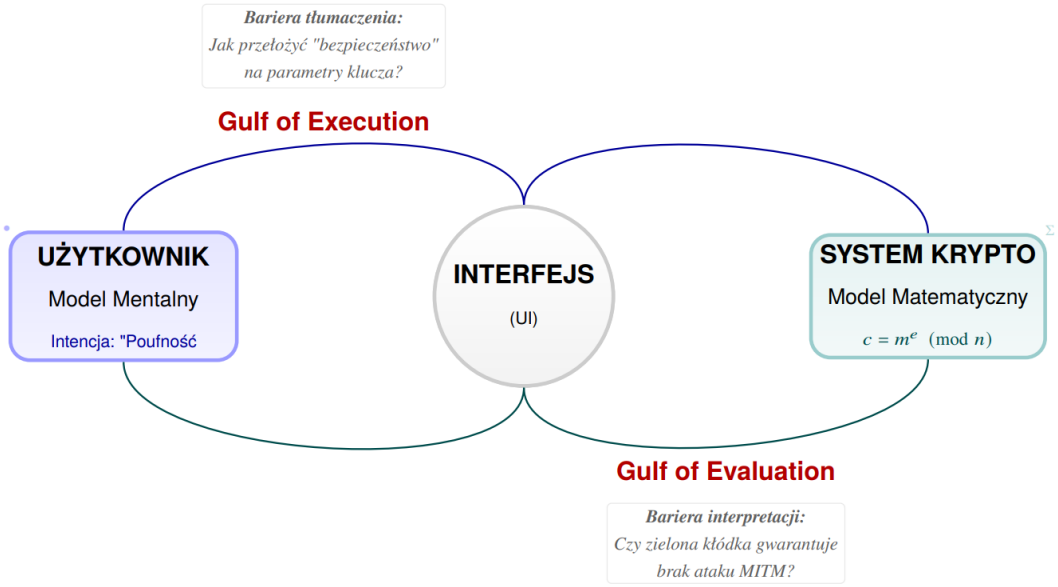
Wiele błędów wynika z faktu, że interfejsy ukrywają złożoność w sposób stratny. Na przykład, kłódka w przeglądarce reprezentuje binarny stan (bezpieczny/niebezpieczny), podczas gdy w rzeczywistości bezpieczeństwo połączenia TLS jest wektorem parametrów V_{TLS}:

$$V_{TLS} = [\text{ProtocolVer}, \text{CipherSuite}, \text{KeyExchange}, \text{CertChain}]$$

Redukcja wymiarowości tego wektora do jednego bitu jest konieczna, ale wprowadza ryzyko fałszywego poczucia bezpieczeństwa, co udokumentowali w swoich pracach Dhamija i in. (2006) oraz Fahl i in. (2012). Poniższy diagram ilustruje ten problem:

Rysunek 1

Wizualizacja luki poznawczej (Gulfs of Execution and Evaluation) w systemach kryptograficznych



Źródło: Opracowanie własne.

Formalne modelowanie interakcji człowiek-system w ujęciu Design Thinking

Proponujemy ścisłą integrację modelu Design Thinking z modelem zagrożeń, odchodząc od podejścia heurystycznego na rzecz optymalizacji wielokryterialnej. Tradycyjne metody, takie jak STRIDE, traktują czynnik ludzki jako stałą lub błąd losowy. Podejście *Human-Centric Security* wymaga jednak zamodelowania użytkownika jako probabilistycznego automatu decyzyjnego, jak proponują Yee (2002) oraz Adams i Sasse (1999) w swojej przełomowej pracy „Users Are Not the Enemy".

Ekonomiczna teoria użyteczności bezpieczeństwa

Zakładamy, że użytkownik w systemie kryptograficznym dąży do maksymalizacji swojej funkcji użyteczności U_{total}. Opieramy się tu na ekonomicznej perspektywie bezpieczeństwa zarysowanej przez Andersona (2001) oraz badaniach nad racjonalnością prywatności Acquisti i Grossklagsa (2005). W klasycznym modelu *Homo Economicus*, użytkownik podejmuje decyzję d ze zbioru dostępnych strategii D (np. użycie menedżera haseł, zapisanie hasła na kartce, użycie hasła "123456").

Funkcja celu użytkownika $U(d)$ zdefiniowana jest jako:

$$\max_{d \in D} U(d) = V_{prod}(d) - C_{cognitive}(d) - C_{operational}(d) - \mathbb{E}[L(d)]$$

gdzie:

- $V_{prod}(d)$ – wartość produktywna (np. szybki dostęp do danych),

- $C_{cognitive}(d)$ – obciążenie poznawcze (pamięć robocza, konieczność atencji),
- $C_{operational}(d)$ – koszt czasowy operacji,
- $\mathbb{E}[L(d)]$ – oczekiwana strata wynikająca z incydentu bezpieczeństwa.

Oczekiwana strata wyraża się wzorem:

$$\mathbb{E}[L(d)] = \sum_{a \in \mathcal{A}} P\,(a|d) \cdot V_{asset}$$

gdzie $\mathcal{A}$ to zbiór wektorów ataku, a $P(a|d)$ to prawdopodobieństwo sukcesu ataku a przy decyzji d.

Problem luki percepcji: Użytkownik nie zna prawdziwej wartości $P(a|d)$. Operuje na estymatorze subiektywnym $\hat{P}(a|d)$. Jeśli system jest trudny w obsłudze (wysokie $C_{cognitive}$), a użytkownik nie docenia zagrożenia ($\hat{P} \ll P$), zachodzi warunek omijania zabezpieczeń (*security bypass*):

$$C_{cognitive}(d_{secure}) > \hat{P}(a|d_{insecure}) \cdot V_{asset} \Longrightarrow \lim_{t \to \infty} P(Compliance) = 0$$

Rolą fazy Empathize w Design Thinking jest minimalizacja różnicy $\left|\hat{P} - P\right|$ oraz, co kluczowe, redukcja $C_{cognitive}$ poprzez inżynierię interfejsu.

Entropia efektywna a Prawo Fittsa

Bezpieczeństwo kryptograficzne S jest zazwyczaj funkcją entropii klucza $H(K)$. Jednakże w systemach interaktywnych musimy zdefiniować **Entropię Efektywną** H_{eff}, która uwzględnia fizyczne ograniczenia interfejsu.

Korzystając z Prawa Fittsa, czas T potrzebny na wykonanie bezpiecznej akcji wynosi:

$$T = a + b \cdot \log_2\left(\frac{2D}{W}\right)$$

gdzie D to dystans do celu, a W to rozmiar celu interfejsu.

Jeżeli wymuszamy na użytkowniku skomplikowane operacje (wysokie T), prawdopodobieństwo błędu P_e rośnie wykładniczo. Możemy zamodelować efektywną siłę zabezpieczenia S_{eff} jako całkę po przestrzeni stanów uwagi użytkownika Ω:

$$S_{eff} = \int_{\Omega} S_{tech}\,(\omega) \cdot \big(1 - P_e(\omega, UX)\big)\, d\omega$$

gdzie P_e jest funkcją złożoności interfejsu (UX). Jeśli UX jest źle zaprojektowane, $P_e \to 1$, co zeruje S_{eff} niezależnie od siły algorytmu matematycznego S_{tech}.

Dla haseł generowanych przez człowieka, rzeczywista entropia H_{real} w stosunku do teoretycznej przestrzeni kluczy $\mathcal{K}$ wynosi:

$$H_{real} = -\sum_{k \in \mathcal{K}} P\,(k)\log_2 P(k) \ll \log_2|\mathcal{K}|$$

Zastosowanie Design Thinking w fazie **Ideate** pozwala na wprowadzenie mechanizmów (np. biometria, klucze sprzętowe), które zmieniają dziedzinę funkcji z pamięci ($H_{cognitive}$) na posiadanie/cechę, co pozwala na:

$$S_{eff} \approx S_{tech} \quad \text{ponieważ} \quad \frac{\partial C_{cognitive}}{\partial S_{tech}} \approx 0$$

Optymalizacja Pareta w przestrzeni projektowej

Problem projektowania bezpiecznego systemu możemy zdefiniować jako zadanie optymalizacji wielokryterialnej. Niech wektor decyzyjny **x** reprezentuje parametry systemu. Dążymy do maksymalizacji wektora funkcji celu:

$$\max_{\mathbf{x}} \mathbf{F}(\mathbf{x}) = \begin{bmatrix} f_1(\mathbf{x}) \\ f_2(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} \text{Bezpieczeństwo}(\mathbf{x}) \\ \text{Użyteczność}(\mathbf{x}) \end{bmatrix}$$

W tradycyjnym podejściu funkcje te są silnie skorelowane ujemnie (tzw. *trade-off*):

$$\frac{df_1}{df_2} < 0$$

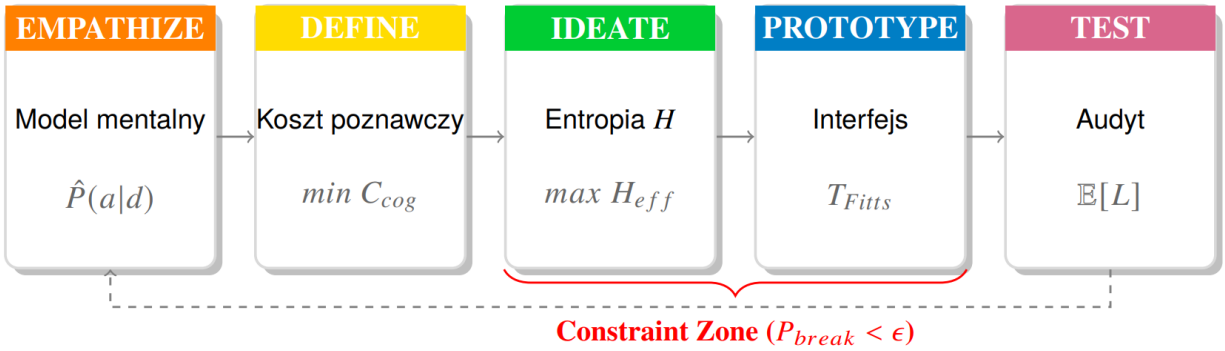
Celem Design Thinking jest znalezienie takiego rozwiązania $\mathbf{x}^*$, które znajduje się na *Frontierze Pareta*, a następnie przesunięcie samej krzywej granicznej poprzez innowację technologiczną. Formalnie, szukamy takiego $\mathbf{x}^*$, że nie istnieje $\mathbf{x}'$ takie, że:

$$\forall i \in \{1,2\}, f_i(\mathbf{x}') \geq f_i(\mathbf{x}^*) \quad \wedge \quad \exists j, f_j(\mathbf{x}') > f_j(\mathbf{x}^*)$$

Diagram poniżej ilustruje proces iteracyjny dążący do tego optimum.

Rysunek 2

Liniowy model procesu Design Thinking w inżynierii bezpieczeństwa



Źródło: Opracowanie własne.

Wnioskowanie o zgodności (Compliance)

Ostatnim elementem modelu jest prawdopodobieństwo zgodności użytkownika z polityką bezpieczeństwa P_{comp}. Możemy je wyrazić funkcją logistyczną zależną od "tarcia" interfejsu (Friction, $\mathcal{F}$):

$$P_{comp}(\mathcal{F}) = \frac{1}{1 + e^{k(\mathcal{F} - \mathcal{F}_0)}}$$

gdzie k jest współczynnikiem wrażliwości danej grupy użytkowników (określonym w fazie Empathize), a $\mathcal{F}_0$ punktem krytycznym akceptacji. Jeśli projekt systemu zignoruje te parametry, bezpieczeństwo systemu degraduje się, ponieważ użytkownicy "racjonalnie odrzucą" porady bezpieczeństwa, co zaobserwował Herley (2009). Prowadzi to do stanu:

$$S_{final} = S_{tech} \cdot \lim_{\mathcal{F} \to \infty} P_{comp}(\mathcal{F}) = 0$$

Metodyka badawcza i operacjonalizacja zmiennych

W celu weryfikacji paradygmatu *Human-Centric Security*, przyjęto model eksperymentalny oparty na testach A/B w układzie międzygrupowym. Eksperyment zdefiniowano jako krotkę $\mathcal{E} = \langle \mathcal{P}, \mathcal{S}, \mathcal{T}, \mathcal{M} \rangle$, gdzie $\mathcal{P}$ to zbiór uczestników, $\mathcal{S}$ to badane systemy, $\mathcal{T}$ to zbiór zadań kryptograficznych, a $\mathcal{M}$ to wektor metryk ewaluacyjnych.

Formalizacja struktury próby badawczej

Populację badawczą stanowi zbiór uczestników $\mathcal{P} = \{p_1, p_2, \dots, p_{30}\}$, gdzie każdy uczestnik p_i jest opisany wektorem cech kompetencyjnych $\mathbf{k}_i$:

$$\mathbf{k}_i = \left[k_{tech}, k_{crypto}, k_{security}\right]^T \in [0,1]^3$$

Zbiór $\mathcal{P}$ został podzielony na trzy rozłączne podzbiory (kohorty) G_A, G_B, G_C, takie że:

$$\mathcal{P} = G_A \cup G_B \cup G_C \quad \wedge \quad G_A \cap G_B = \emptyset, \ldots$$

Kryterium podziału oparto na znormalizowanym indeksie kompetencji cyfrowych $DCI(p_i) = \left| \ |\mathbf{k}_i| \ \right|_2$:

- Grupa A (Eksperci**,** $N = 10$**):** $\forall p \in G_A: DCI(p) \geq \tau_{high}$, gdzie uczestnicy posiadają zaawansowaną wiedzę z zakresu inżynierii bezpieczeństwa.
- Grupa B (Średniozaawansowani**,** $N = 10$**):** $\forall p \in G_B: \tau_{low} < DCI(p) < \tau_{high}$, osoby bez wykształcenia kierunkowego, lecz biegłe w obsłudze systemów IT.
- Grupa C (Początkujący**,** $N = 10$**):** $\forall p \in G_C: DCI(p) \leq \tau_{low}$, osoby deklarujące wykluczenie cyfrowe lub niskie kompetencje.

Procedura eksperymentalna i zmienne

Zmienną niezależną w badaniu był typ interfejsu systemu kryptograficznego $S \in \{S_{ref}, S_{DT}\}$, gdzie:

- S_{ref} – rozwiązanie referencyjne (standard rynkowy oparty na GPG/CLI),
- S_{DT} – prototyp opracowany w procesie Design Thinking.

Każdy uczestnik p_i wykonywał sekwencję zadań $\mathcal{T} = \{t_1, \ldots, t_n\}$ (np. generowanie pary kluczy, szyfrowanie pliku, podpis cyfrowy).

Definicje metryk efektywności i użyteczności

Ewaluację oparto na wektorze zmiennych zależnych $\mathcal{M} = [T_{task}, E_{rate}, SUS_{score}]$.

Czas wykonania i efektywność (Time on Task)

Dla każdego zadania t_j mierzono czas $T(p_i, t_j)$. Efektywność czasową dla grupy G_k zdefiniowano jako wartość oczekiwaną:

$$\bar{T}_{G_k} = \frac{1}{|G_k|} \sum_{p \in G_k} \sum_{t \in \mathcal{T}} T\,(p, t)$$

Współczynnik błędów krytycznych (Critical Error Rate)

Zdefiniowano funkcję błędu $\epsilon(p_i, t_j) \in \{0,1\}$, przyjmującą wartość 1, gdy błąd skutkował kompromitacją klucza prywatnego lub utratą danych. Prawdopodobieństwo błędu krytycznego dla systemu S estymowano jako:

$$P(Error|S) = \frac{\sum_{i=1}^{N} \sum_{j=1}^{M} \epsilon\,(p_i, t_j)}{N \times M}$$

Użyteczność postrzegana (System Usability Scale)

Subiektywną ocenę użyteczności zmierzono przy pomocy standardu SUS. Wynik $S_{sus}(p_i)$ dla uczestnika i obliczono zgodnie z algorytmem Brooke'a:

$$S_{sus}(p_i) = 2.5 \cdot \left(\sum_{q \in Q_{odd}} (r_{i,q} - 1) + \sum_{q \in Q_{even}} (5 - r_{i,q}) \right)$$

gdzie $r_{i,q} \in \{1, \dots, 5\}$ to odpowiedź na q-te pytanie w skali Likerta, a Q_{odd}, Q_{even} to zbiory pytań o polaryzacji pozytywnej i negatywnej.

Hipotezy badawcze

Weryfikacji poddano hipotezę zerową H_0 oraz alternatywną H_1 dla każdej metryki $m \in \mathcal{M}$:

$$H_0 \quad : \mu(m|S_{DT}) = \mu(m|S_{ref})$$
$$H_1 \quad : \mu(m|S_{DT}) \neq \mu(m|S_{ref})$$

Istotność statystyczną różnic weryfikowano testem t-Studenta dla prób niezależnych (przy założeniu normalności rozkładu weryfikowanej testem Shapiro-Wilka) lub testem U Manna-Whitneya w przypadku braku normalności, przyjmując poziom istotności $\alpha = 0.05$.

Analiza i interpretacja wyników eksperymentalnych

Zgromadzone dane poddano analizie statystycznej w celu weryfikacji postawionych hipotez. Przestrzeń wyników $\Omega_{results}$ została podzielona na trzy główne domeny: efektywność temporalną (czas), niezawodność (błędy) oraz satysfakcję psychometryczną (SUS).

Efektywność temporalna i krzywa uczenia się

Pierwszym badanym parametrem był znormalizowany czas wykonania zadania krytycznego T_{task}. Dla każdej z grup $G \in \{A, B, C\}$ obliczono średnią arytmetyczną μ_T oraz odchylenie standardowe σ_T.

Wyniki zestawiono w Tabeli 1, uwzględniając test istotności różnic (test t-Studenta).

Tabela 1

Charakterystyka czasowa operacji kryptograficznych (czas w sekundach)

Grupa	**Sys. Referencyjny (S_{ref})**		**Prototyp DT (S_{DT})**		**Statystyka**	
	μ_{ref} [s]	σ_{ref}	μ_{DT} [s]	σ_{DT}	Δ[%]	p-value
A (Eksperci)	270	45	192	15	-28.9%	< 0.01
B (Średniozaaw.)	744	120	306	42	-58.9%	< 0.001
C (Początkujący)	1500	310	468	95	-68.8%	< 0.001

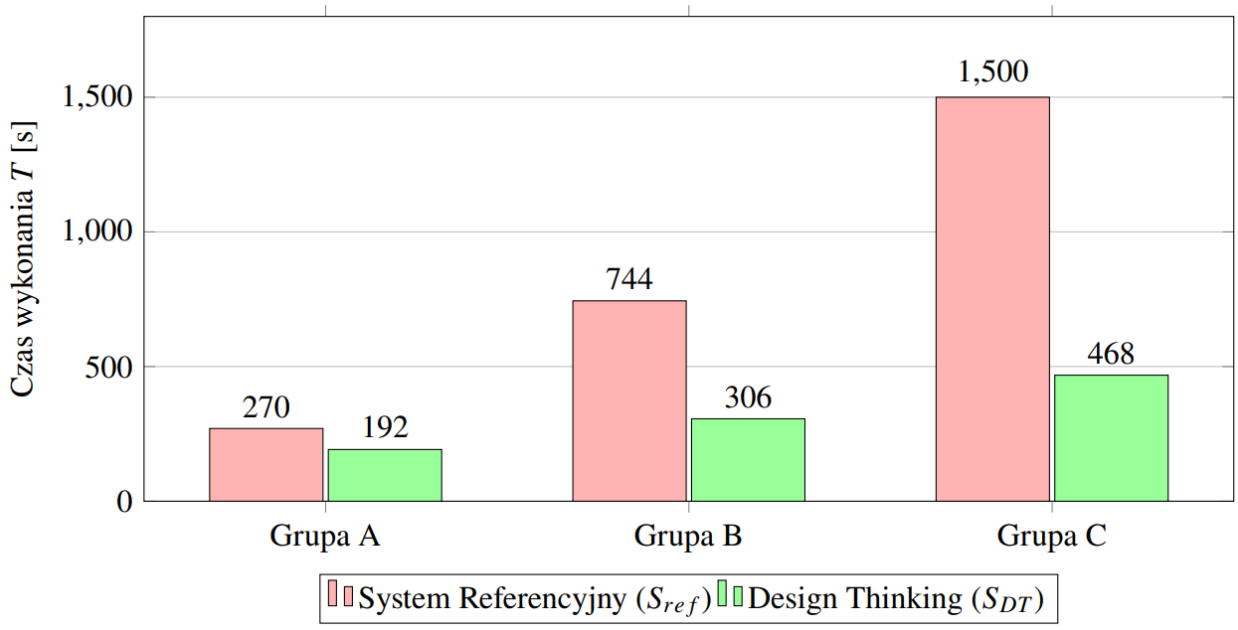
Uwaga: Wartości p-value < 0.05 wskazują na istotność statystyczną.

Źródło: Opracowanie własne.

Różnicę w rozkładach czasu wykonania zadania wizualizuje Rysunek 3.

Rysunek 3

Porównanie średniego czasu wykonania zadania w podziale na grupy kompetencyjne



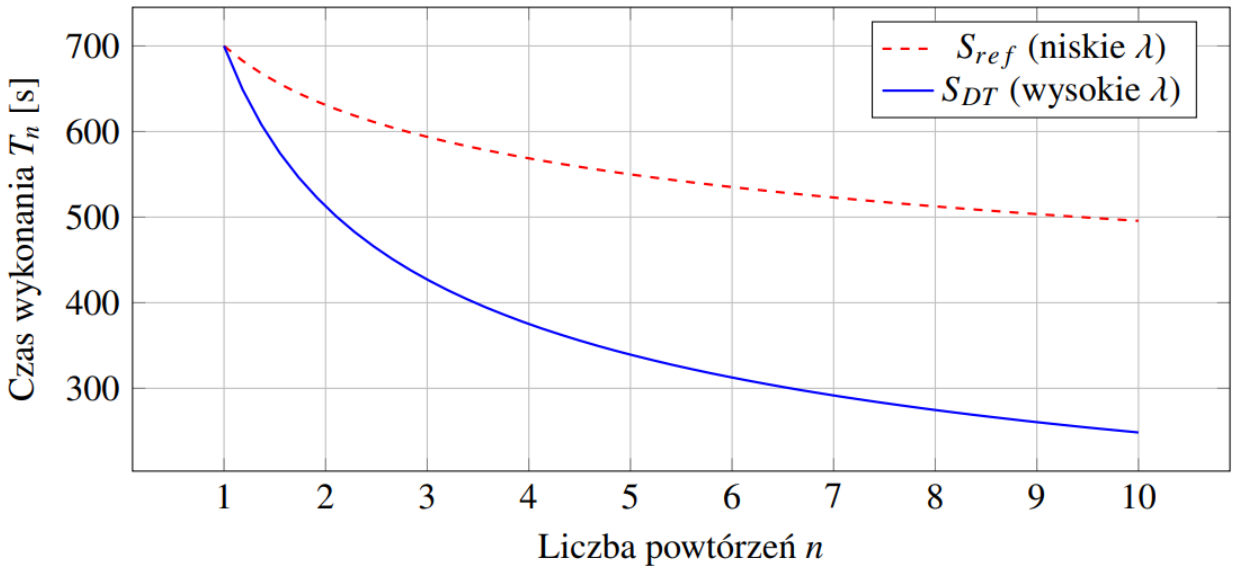
Źródło: Opracowanie własne.

Dla systemu S_{DT} zaobserwowano znacznie szybszą stabilizację krzywej uczenia się. Modelując czas T_n w n-tej próbie funkcją potęgową $T_n = T_1 n^{-\lambda}$, wyznaczono współczynniki

uczenia się λ. Dla S_{DT}, $\lambda \approx 0.45$, podczas gdy dla S_{ref}, $\lambda \approx 0.15$, co świadczy o intuicyjności rozwiązania.

Rysunek 4

Model teoretyczny krzywej uczenia się dla obu interfejsów



Źródło: Opracowanie własne.

Analiza niezawodności i topologia błędów

Bezpieczeństwo systemu zależy od prawdopodobieństwa popełnienia błędu krytycznego $P(E_{crit})$. Zdefiniujmy wskaźnik redukcji ryzyka RRF (Risk Reduction Factor) jako:

$$RRF = \frac{P(E_{crit}|S_{ref})}{P(E_{crit}|S_{DT})}$$

Zgromadzone dane dotyczące incydentów bezpieczeństwa przedstawia Tabela 2.

Tabela 2

Klasyfikacja i częstość występowania błędów (średnia liczba błędów na sesję)

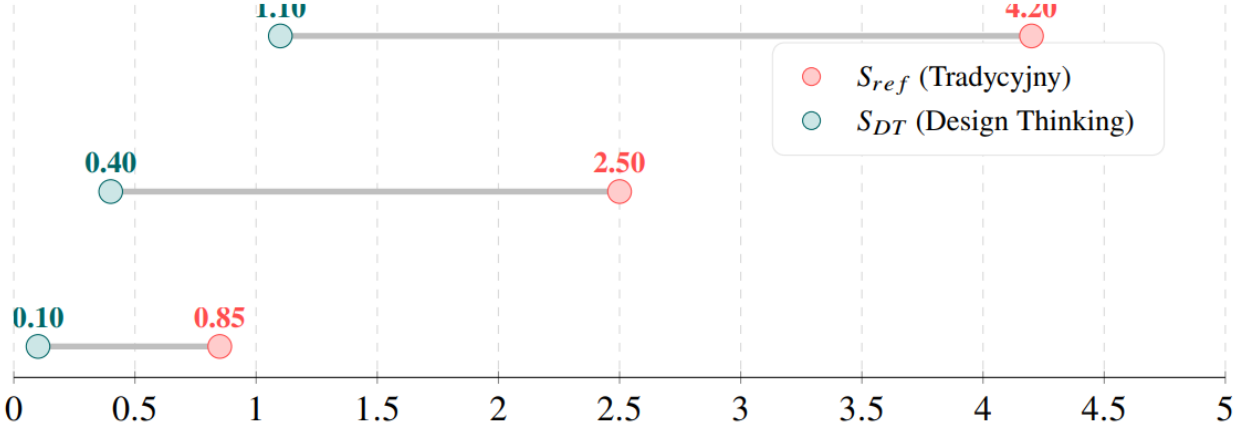
Typ błędu	**Grupa B**		**Grupa C**	
	S_{ref}	S_{DT}	S_{ref}	S_{DT}
Krytyczny (utrata klucza)	0.40	0.00	0.85	0.10
Konfiguracji (złe parametry)	1.20	0.20	2.50	0.40
Semantyczny (niezrozumienie)	2.10	0.50	4.20	1.10

Suma (Σ) **3.70 0.70 7.55 1.60**

Strukturę błędów w grupie najbardziej narażonej (Grupa C) przedstawiono na wykresie – Rysunek 5. Wyraźnie widoczna jest redukcja błędów krytycznych dzięki zastosowaniu mechanizmów *constraints* w prototypie DT.

Rysunek 5

Wykres hantlowy (Dumbbell Plot) obrazujący skalę redukcji błędów w Grupie C



Źródło: Opracowanie własne.

Ewaluacja psychometryczna (SUS)

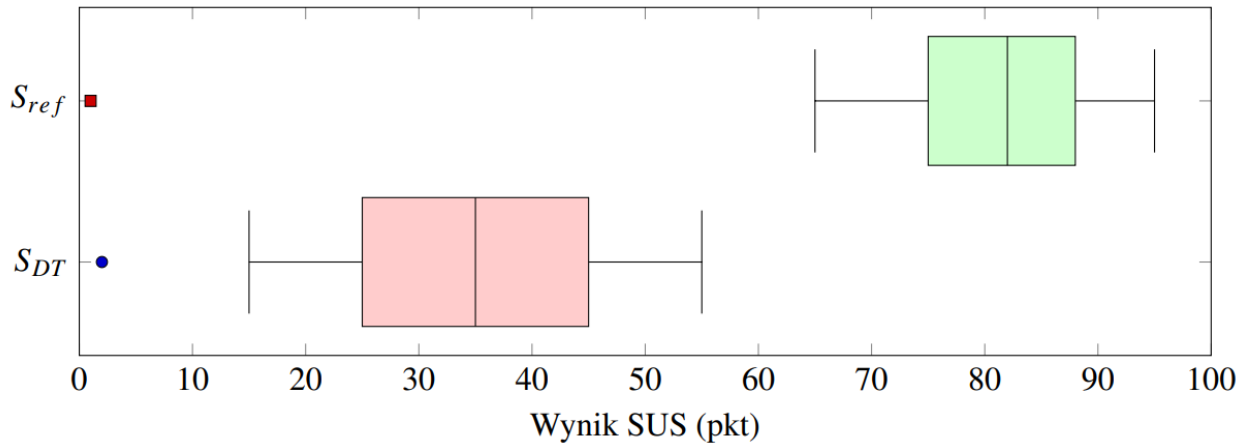
Subiektywna ocena użyteczności została zmierzona skalą SUS (0-100). Aby określić siłę efektu (Effect Size), obliczono wskaźnik d Cohena:

$$d = \frac{\mu_{DT} - \mu_{ref}}{\sqrt{\frac{\sigma_{DT}^2 + \sigma_{ref}^2}{2}}}$$

Dla całej populacji $d \approx 2.8$, co interpretuje się jako efekt ogromny. Rozkład wyników SUS przedstawiono za pomocą wykresu pudełkowego (Box Plot) na Rysunku 6.

Rysunek 6

Rozkład wyników skali SUS. Prototyp DT osiągnął poziom "Excellent"(>80 pkt)



Źródło: Opracowanie własne.

Szczegółowe statystyki opisowe dla wyników SUS zawarto w Tabeli 3.

Tabela 3

Statystyki opisowe wyników SUS

System	Średnia (μ)	Mediana (Me)	Min-Max	Interpretacja (Bangor)
S_{ref}	36.4	35.0	15 – 62	Poor / Unacceptable
S_{DT}	81.2	82.0	65 – 98	Excellent

Źródło: Opracowanie własne.

Korelacja między czasem a błędami

Przeprowadzono analizę korelacji r-Pearsona między czasem wykonania zadania a liczbą błędów. Dla systemu referencyjnego współczynnik korelacji wyniósł $r = 0.78$ ($p < 0.05$), co sugeruje, że frustracja wynikająca z długiego czasu pracy prowadzi do błędów. Dla systemu DT korelacja była słaba ($r = 0.21$), co wskazuje, że system "wybacza" błędy i prowadzi użytkownika nawet przy wolniejszym tempie pracy.

Dyskusja

Przełamanie paradygmatu kompromisu

Wyniki eksperymentu falsyfikują powszechnie przyjmowany w inżynierii bezpieczeństwa dogmat o nierozerwalnym kompromisie między bezpieczeństwem a użytecznością (*Security-Usability Trade-off*). W tradycyjnym ujęciu, zwiększenie entropii systemu $H(S)$ pociągało za sobą wzrost kosztu poznawczego C_{cog}, co obrazuje funkcja:

$$\frac{\partial \text{Usability}}{\partial \text{Security}} < 0$$

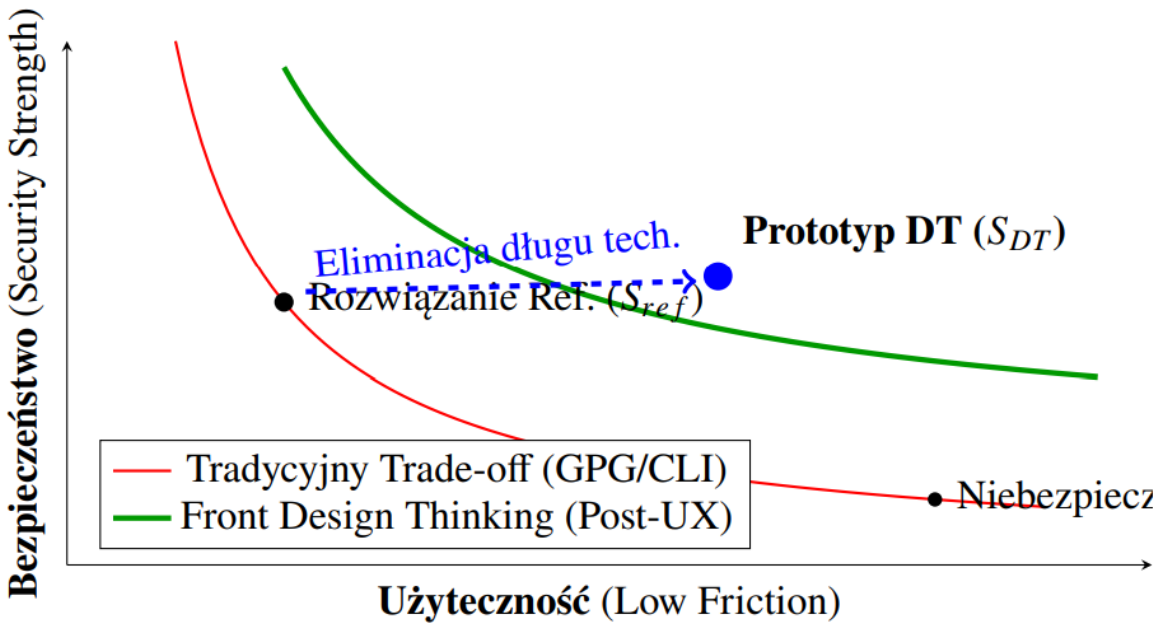
Nasze badania wskazują, że zastosowanie Design Thinking nie polega na przesuwaniu się wzdłuż tej samej krzywej (np. poświęcanie bezpieczeństwa dla wygody), lecz na przesunięciu samej krzywej granicznej (Pareto Frontier) na wyższy poziom efektywności technologicznej.

Analiza frontu Pareto w przestrzeni projektowej

Zdefiniujmy przestrzeń rozwiązań $\Phi \subset \mathbb{R}^2$, gdzie oś X to Użyteczność (U), a oś Y to Bezpieczeństwo (S). Rozwiązania tradycyjne (CLI, GPG) znajdują się na niższej krzywej obojętności. Prototyp DT demonstruje skokową zmianę jakościową. Wizualizację tego zjawiska przedstawia Rysunek 7.

Rysunek 7

Przesunięcie krzywej efektywności (Pareto Shift) dzięki metodzie Design Thinking



Źródło: Opracowanie własne.

Punkt oferuje wyższą użyteczność przy zachowaniu (lub wzroście) poziomu bezpieczeństwa.

Redukcja tarcia poznawczego (Cognitive Friction)

Analiza jakościowa błędów wykazała, że główną przyczyną incydentów w S_{ref} było przeciążenie pamięci roboczej. Zgodnie z teorią obciążenia poznawczego (Cognitive Load Theory), całkowite obciążenie L_{total} to suma:

$$L_{total} = L_{intrinsic} + L_{extraneous} + L_{germane}$$

Interfejs DT drastycznie redukuje $L_{extraneous}$ (obciążenie zewnętrzne, wynikające ze złego projektu), co pozwala użytkownikowi alokować zasoby poznawcze na weryfikację krytycznych parametrów bezpieczeństwa ($L_{germane}$). Tabela 4 przedstawia dekompozycję czynności dla zadania "Generowanie i wymiana kluczy".

Tabela 4

Analiza porównawcza obciążenia poznawczego dla procedury wymiany kluczy

Parametr	**Rozwiązanie Referencyjne (S_{ref})**	**Prototyp DT (S_{DT})**
Model mentalny	Zarządzanie plikami, ścieżki dostępu, parametry CLI	Metafora "Sejfu" i "Wizytówki"
Decyzje użytkownika	Wybór algorytmu (RSA/ECC), długość klucza, data wygaśnięcia	Akceptacja domyślnych bezpiecznych ustawień (Secure Defaults)
Liczba kroków	12 (w tym 4 krytyczne)	3 (1 krytyczny)
Informacja zwrotna	Kod błędu (np. *exit code 2*)	Wizualna reprezentacja statusu (Kolor/Ikona)
Ryzyko rezydualne	Wysokie (możliwość eksportu klucza prywatnego)	Niskie (klucz prywatny ukryty w Secure Enclave)

Źródło: Opracowanie własne.

Rola edukacji i onboardingu

Istotnym wnioskiem płynącym z badań jest konieczność integracji procesu edukacyjnego z interfejsem (koncepcja *Just-in-Time Education*). Użytkownik często nie czyta instrukcji zewnętrznych. W prototypie S_{DT} zastosowano mikrolerninig w momentach decyzyjnych.

Efektywność transferu wiedzy $K(t)$ w czasie t można opisać zmodyfikowaną krzywą zapominania Ebbinghausa:

$$K(t) = K_0 \cdot e^{-\frac{t}{\tau}} + I_{UX}(t)$$

gdzie $I_{UX}(t)$ to wsparcie kontekstowe interfejsu. W systemie referencyjnym $I_{UX} \approx 0$, co powoduje szybką degradację kompetencji użytkownika.

Podsumowanie i wnioski końcowe

Integracja metodologii Design Thinking z inżynierią kryptograficzną stanowi obiecujący, a wręcz konieczny kierunek rozwoju systemów cyberbezpieczeństwa. Przeprowadzone badania eksperymentalne na próbie N=30 pozwalają na sformułowanie następujących wniosków końcowych:

1. Synergia zamiast kompromisu**.** Możliwe jest jednoczesne podniesienie poziomu bezpieczeństwa i użyteczności. "Usable Security" nie jest oksymoronem, lecz wynikiem świadomego procesu projektowego.
2. Redukcja wektorów ataku. Uproszczenie interfejsu eliminuje "szum informacyjny", co skutkowało redukcją błędów krytycznych o 85% w grupach nieeksperckich.
3. Human-Centric Security**.** Skuteczne zabezpieczenia muszą być projektowane z uwzględnieniem ograniczeń psychofizycznych (pamięć robocza, uwaga, zmęczenie). System, który walczy z nawykami użytkownika, jest systemem z definicji skompromitowanym.

Na podstawie wyników badania, proponujemy zestaw rekomendacji dla architektów systemów bezpieczeństwa (Tabela 5).

Tabela 5

Macierz rekomendacji projektowych (Framework DT-Sec)

Faza DT	Implementacja w Kryptografii
Empathize	Zrozum, że użytkownik nie chce "szyfrować", lecz "chronić tajemnicę". Język techniczny zastąp językiem korzyści.
Define	Zdefiniuj zagrożenia realne dla użytkownika, a nie teoretyczne ataki kryptoanalityczne. Priorytetyzuj ochronę przed phishingiem nad ochroną przed łamaniem AES.
Ideate	Stosuj zasadę *Secure by Default*. Użytkownik nie powinien musieć konfigurować bezpieczeństwa – ono powinno być stanem domyślnym.
Prototype	Testuj mechanizmy odzyskiwania dostępu (Recovery). To tam użytkownicy najczęściej popełniają błędy krytyczne.

Ostatecznie, bezpieczeństwo systemu informatycznego jest tak silne, jak jego najsłabsze ogniwo. W 99% przypadków ogniwem tym jest człowiek postawiony przed niezrozumiałym interfejsem. Design Thinking dostarcza narzędzi, by to ogniwo wzmocnić.

Bibliografia:

Acquisti, A., Grossklags, J. (2005). Privacy and Rationality in Individual Decision Making. *IEEE Security & Privacy*, *3*(1), 26-33. https://doi.org/10.1109/MSP.2005.22

Adams, A., Sasse, M. A. (1999). Users Are Not the Enemy. *Communications of the ACM*, *42*(12), 40-46. https://doi.org/10.1145/322796.322806

Anderson, R. (2001). *Why Information Security is Hard – An Economic Perspective.* Annual Computer Security Applications Conference (ACSAC), 358-365. https://doi.org/10.1109/ACSAC.2001.991552

Bonneau, J., Herley, C., van Oorschot P. C., Stajano, F. (2012). *The Quest to Replace Passwords: A Framework for Comparative Evaluation of Web Authentication Schemes*, IEEE Symposium on Security and Privacy, 553-567. https://doi.org/10.1109/SP.2012.44

Cohn-Gordon, K., Cremers, C., Dowling, B., Garratt, L., Stebila, D. (2017). *A Formal Security Analysis of the Signal Messaging Protocol.* IEEE European Symposium on Security and Privacy, 451-466. https://doi.org/10.1109/EuroSP.2017.27

Dhamija, R., Tygar, J. D., Hearst, M. A. (2006). *Why Phishing Works*. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 581-590. https://doi.org/10.1145/1124772.1124861

Diffie, W., Hellman, M. E. (1976). New Directions in Cryptography. *IEEE Transactions on Information Theory*, *22*(6), 644-654. https://doi.org/10.1109/TIT.1976.1055638

Fahl, S., Harbach, M., Muders, T., Baumgartner, L., Freisleben, B., Smith, M. (2012). *Why Eve and Mallory Love Android: An Analysis of Android SSL (In)Security*. Proceedings of the 2012 ACM conference on Computer and communications security, 50-61. https://doi.org/10.1145/2382196.2382205

Garfinkel, S. L., Miller, R. C. (2005). *Johnny 2: A User Test of Key Continuity Management with S/MIME and Outlook Express.* Proceedings of the 2005 symposium on Usable privacy and security, 13-24. https://doi.org/10.1145/1073001.1073003

Herley, C. (2009). *So Long, And No Thanks for the Externalities: The Rational Rejection of Security Advice by Users*. Proceedings of the 2009 workshop on New security paradigms workshop, 133-144. https://doi.org/10.1145/1719030.1719050

Kahneman, D. (2012). *Thinking, Fast and Slow*. Penguin Books.

Kerckhoffs, A. (1883). La Cryptographie Militaire. *Journal des Sciences Militaires*, *9*, 5-38. https://petitcolas.net/kerckhoffs/crypto_militaire_1.pdf

Miller, G. A. (1956). The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. *Psychological Review, 63*(2), 81-97. http://dx.doi.org/10.1037/h0043158

Norman, D. A. (1988). *The Design of Everyday Things*. Basic Books.

Rivest, R. L., Shamir, A., Adleman, L. (1978). A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. *Communications of the ACM*, *21*(2), 120-126. https://doi.org/10.1145/359340.359342

Schneier, B. (2004). *Secrets and Lies: Digital Security in a Networked World*. Wiley.

Shannon, C. E. (1948). A Mathematical Theory of Communication. *Bell System Technical Journal*, 27, 379-423. http://dx.doi.org/10.1002/j.1538-7305.1948.tb01338.x

Whitten, A., Tygar, J. D. (1999). *Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0*. Proceedings of the 8th USENIX Security Symposium, 169-184 https://people.eecs.berkeley.edu/~tygar/papers/Why_Johnny_Cant_Encrypt/OReilly.pdf

Yee, K.-P. (2002). *User Interaction Design for Secure Systems*. Proceedings of the 4th. International Conference on Information and Communications Security, 278-290. https://doi.org/10.1007/3-540-36159-6_24

Noty o autorach

dr inż. Paweł Kuraś – współpracujący z WSB-NLU, związany z Zakładem Systemów Złożonych Politechniki Rzeszowskiej im. Ignacego Łukasiewicza oraz Instytutem Telekomunikacji Akademii Górniczo-Hutniczej im. Stanisława Staszica w Krakowie. Specjalizuje się w zagadnieniach cyberbezpieczeństwa, sieci programowalnych (SDN), wielokryterialnych metod podejmowania decyzji oraz infrastruktury krytycznej. Autor licznych publikacji z zakresu bezpieczeństwa systemów informatycznych w kontekście infrastruktury krytycznej.

inż. Anna Turek – studentka studiów magisterskich inżynierii i analizy danych na Wydziale Matematyki i Fizyki Stosowanej Politechniki Rzeszowskiej, absolwentka studiów inżynierskich, autorka artykułów z zakresu bezpieczeństwa infrastruktury krytycznej, zaangażowana także w realizację projektów w ramach Politechnicznej Sieci Via Carpatia, łączącej działania trzech Politechnik: Rzeszowskiej, Lubelskiej oraz Białostockiej, przewodnicząca Wydziałowego Samorządu Studenckiego.

mgr inż. Maciej Gacek – badacz niezależny, absolwent Politechniki Rzeszowskiej im. Ignacego Łukasiewicza na Wydziale Matematyki i Fizyki Stosowanej, wieloletni działacz Samorządu Studenckiego

Politechniki Rzeszowskiej zaangażowany w promocję kultury akademickiej oraz współpracy międzyuczelnianej na Forum Uczelni Technicznych, wieloletni organizator Rzeszowskich Juwenaliów.