BEZPEČNOSTNÍ REPORT

Bezpečnostní kontrola webové aplikace

Klient: Scheduly s.r.o. (fiktivní ukázkový klient)

Testovaný systém: app.scheduly.example — rezervační a fakturační SaaS

Období testu: 3.–5. 6. 2026 (ilustrativní)

Verze reportu: 1.0

Klasifikace dokumentu: Ukázkový dokument (veřejná ukázka)

Zpracoval: Milan Půlkrábek, Digitální Sebeobrana

> Toto je ukázkový report. Klient, aplikace i nálezy jsou smyšlené a slouží pouze k demonstraci struktury a úrovně výstupu Bezpečnostní kontroly webové aplikace. Nejde o reálně testovaný systém ani o referenci konkrétního klienta.

1. Shrnutí pro rozhodnutí

Testované uživatelské toky byly během kontroly dostupné a bylo možné je dokončit. Kontrola v potvrzeném rozsahu ale našla závažné nedostatky v řízení přístupu: aplikace důsledně neověřuje, komu záznam patří a kdo ho smí měnit. Ve výchozím stavu si přihlášený uživatel může přečíst faktury cizích účtů (F-01) a povýšit se na administrátora (F-02). Tyto dva nálezy mají nejvyšší prioritu k nápravě.

Celkové hodnocení

Oblast	Hodnocení / komentář
Celkový stav	Testované toky jsou funkční, aplikace však obsahuje závažné nedostatky v řízení přístupu; vyžaduje přednostní nápravu.
Nejdůležitější riziko	Přístup k datům cizích účtů a možnost eskalace oprávnění na administrátora.
Doporučená priorita	Přednostně opravit nálezy F-01 a F-02 (řízení přístupu).
Nutný další krok	Oprava kritických nálezů F-01 a F-02, poté cílený retest dotčených toků.

Klíčová doporučení

- Kontrolovat vlastnictví každého záznamu (faktury, klienti, nastavení účtu) na straně serveru, ne jen skrývat na frontendu.
- Odebrat citlivá pole (role, oprávnění) z dat přijímaných od klienta a role řídit výhradně na straně serveru.
- Doplnit omezení pokusů o přihlášení a sjednotit chybové hlášky, aby nešlo zjišťovat existenci účtů.

Poznámka: Hodnocení platí pouze pro domluvený rozsah a stav systému v době testu.

2. Rozsah testu

Položka	Potvrzený rozsah
Testovaná aplikace / doména	app.scheduly.example (staging prostředí)
Prostředí	TEST — staging; podle informací klienta konfiguračně odpovídá produkci
Uživatelské role	Neregistrovaný návštěvník; Uživatel (majitel účtu); Administrátor platformy
Hlavní funkce a scénáře	Registrace a přihlášení; správa klientů; vytvoření a odeslání faktury; veřejný odkaz na fakturu; nastavení účtu a API klíč
Relevantní rozhraní	Webové rozhraní a REST API (/api/*)
Části mimo rozsah	Platební brána třetí strany; e-mailová infrastruktura; síťová vrstva a DDoS; mobilní aplikace
Provozní omezení	Bez destruktivních akcí a bez zátěžového testování; testováno po dohodě v pracovní době

Cíl testu

Nezávisle ověřit bezpečnost aplikace v potvrzeném rozsahu se zaměřením na řízení přístupu, autentizaci a práci s daty a dodat prioritizovaná doporučení k nápravě.

3. Přístup a metodika

Kontrola kombinovala ruční ověřování s podpůrnými nástroji. Testování se zaměřilo na skutečné chování aplikace v potvrzeném rozsahu, nikoli pouze na automatický výpis potenciálních problémů.

- Mapování dostupných funkcí a testovaných rolí
- Ověření přihlášení, relací a řízení přístupu
- Testování práce se vstupy, daty a relevantními aplikačními toky
- Ruční ověření a posouzení praktického dopadu nálezů
- Prioritizace nápravy podle dopadu a zneužitelnosti

Rozsah pokrytí

Oblast	Co bylo ověřeno	Ověření
Autentizace a relace	Přihlášení, odhlášení, atributy a životnost session cookie, reset hesla	Ověřeno
Řízení přístupu	Horizontální (mezi účty) i vertikální (role) autorizace na webu i API	Ověřeno
Vstupy a validace	Validace klíčových endpointů a chování při neplatných typech	Ověřeno
Business logika toků	Vytvoření a odeslání faktury, veřejný odkaz, správa klientů	Ověřeno
Konfigurace a hlavičky	Bezpečnostní hlavičky, chybové odpovědi, vynucení TLS	Ověřeno

„Ověřeno“ znamená prověření v potvrzených tocích a rozsahu této kontroly, nikoli vyčerpávající prověření všech endpointů a variant dané oblasti.

Tato veřejná ukázka používá textové záznamy požadavků a odpovědí; konkrétní podoba důkazu (výřez requestu, snímek obrazovky, log) se u ostrých zakázek volí podle nálezu a rozsahu.

4. Omezení a předpoklady

Testování proběhlo pouze v potvrzeném rozsahu.

Výsledky odpovídají stavu systému v době testu.

Nebyly prováděny zakázané ani provozně rizikové akce uvedené ve scopu.

Test probíhal na staging prostředí; drobné rozdíly oproti produkci nelze vyloučit.

Platební brána třetí strany je mimo rozsah kontroly (viz Rozsah testu) a nebyla testována vůbec, včetně samotného předání.

5. Přehled nálezů

ID	Název	Závažnost	Stav	Doporučená priorita
F-01	Čtení faktur cizích účtů (IDOR)	KRITICKÁ	OTEVŘENO	1
F-02	Eskalace na administrátora přes profil (mass assignment)	KRITICKÁ	OTEVŘENO	1
F-03	Nedostatečná ochrana proti automatizovaným pokusům o přihlášení	STŘEDNÍ	OTEVŘENO	2
F-04	Zjistitelnost existence účtů (user enumeration)	STŘEDNÍ	OTEVŘENO	3
F-05	Detailní chybové odpovědi	NÍZKÁ	OTEVŘENO	3

6. Detail nálezů

F-01 Čtení faktur cizích účtů (IDOR)

Pole	Obsah
Závažnost	KRITICKÁ
Stav	OTEVŘENO
Dotčená část	GET /api/invoices/{id}
Kategorie	Řízení přístupu (IDOR)
OWASP / CWE	A01:2025 Broken Access Control · CWE-639
Důvěra v nález	Potvrzeno

Popis

Endpoint pro detail faktury ověřuje pouze to, že je uživatel přihlášen, nikoli že faktura patří jeho účtu. Změnou číselného ID v adrese lze zobrazit faktury jiných účtů.

Praktický dopad

Přihlášený uživatel může systematicky číst faktury cizích firem — jména klientů, fakturované částky, adresy i popis plnění. Ověřeno na dvou testovacích účtech: účet A zobrazil fakturu účtu B. Jde o únik osobních a

obchodních údajů napříč zákazníky.

Důkaz a reprodukce

Předpoklady: dva běžné účty (A, B), každý s vlastní fakturou

Krok 1: jako účet B otevřít vlastní fakturu a poznamenat její ID (např. 1042)

Krok 2: jako účet A zavolat GET /api/invoices/1042

Výsledek: API vrátí kompletní fakturu účtu B (HTTP 200) bez kontroly vlastnictví

Zachycený průběh:

```
# 1) Přihlášení jako účet A (user_id 31), uložení session
$ curl -s -c a.txt -X POST https://app.scheduly.example/api/auth/login \
       -H 'Content-Type: application/json' \
       -d '{"email":"user-a@scheduly.example","password":"***"}'
{"status":"ok","user_id":31}

# 2) Účet A si vyžádá fakturu 1042, která patří účtu B (account_id 58)
$ curl -i -s -b a.txt https://app.scheduly.example/api/invoices/1042
HTTP/1.1 200 OK
Content-Type: application/json
{
  "id": 1042,
  "account_id": 58,
  "client_name": "Demo klient B",
  "amount": 48250, "currency": "CZK",
  "items": [ ... ]
}

# Účet A (31) čte fakturu cizího účtu (58). Server neověřuje vlastnictví.
```

Doporučení k nápravě

U každého přístupu k záznamu ověřovat na straně serveru, že přihlášený uživatel je jeho vlastníkem, případně má explicitní oprávnění. Kontrolu vlastnictví aplikovat centrálně na všechny endpointy /api/* pracující s ID záznamu, ne pouze na faktury. Očekávaný stav: cizí ID vrací 403/404 bez jakýchkoli dat.

Ověření opravy

Retest s účty A a B na fakturách (endpoint dotčený tímto nálezem); cizí ID nesmí vrátit žádná data.

F-02 Eskalace na administrátora přes profil (mass assignment)

Pole	Obsah
Závažnost	KRITICKÁ
Stav	OTEVŘENO
Dotčená část	PATCH /api/users/me
Kategorie	Řízení přístupu / mass assignment
OWASP / CWE	A01:2025 Broken Access Control · CWE-915
Důvěra v nález	Potvrzeno

Popis

Endpoint pro úpravu vlastního profilu přebírá celé tělo požadavku a ukládá i pole „role“. Uživatel může do požadavku přidat role: "admin" a nastavit si administrátorská oprávnění.

Praktický dopad

Kterýkoli registrovaný uživatel se může povýšit na administrátora platformy a získat přístup k datům a nastavení

všech účtů. V kombinaci s F-01 to znamená úplné prolomení oddělení dat mezi zákazníky.

Důkaz a reprodukce

Předpoklady: běžný registrovaný účet

Krok 1: odeslat PATCH /api/users/me s tělem obsahujícím "role":"admin"

Krok 2: znovu načíst profil a administrátorské rozhraní

Výsledek: účet získá roli admin a přístup k administrátorským funkcím

Zachycený průběh:

```
# Běžný uživatel (role "user") upraví profil a přidá pole "role"
$ curl -s -b a.txt -X PATCH https://app.scheduly.example/api/users/me \
       -H 'Content-Type: application/json' \
       -d '{"name":"Alice","role":"admin"}'
{"id":31,"name":"Alice","role":"admin"}

# Dříve zamítnutý administrátorský endpoint je nyní dostupný
$ curl -s -b a.txt -o /dev/null -w "%{http_code}\n" \
       https://app.scheduly.example/api/admin/users
200
```

Doporučení k nápravě

Na straně serveru zpracovávat jen explicitně povolená pole (whitelist) — jméno, e-mail apod. Pole jako „role“ z profilového požadavku explicitně odmítnout a pokus zalogovat, rozhodně ho nepromítat do dat. Role a oprávnění měnit jen odděleným, autorizovaným tokem. Očekávaný stav: pole „role“ z profilového požadavku nemá žádný účinek.

Ověření opravy

Pokus o změnu role přes profil nesmí mít efekt; ověřit i ostatní zapisovací endpointy na stejný vzor.

F-03 Nedostatečná ochrana proti automatizovaným pokusům o přihlášení

Pole	Obsah
Závažnost	STŘEDNÍ
Stav	OTEVŘENO
Dotčená část	POST /api/auth/login
Kategorie	Autentizace
OWASP / CWE	A07:2025 Authentication Failures · CWE-307
Důvěra v nález	Potvrzeno

Popis

V sérii 200 rychlých sekvenčních pokusů o přihlášení s neplatným heslem endpoint nevrátil kód 429 ani jinak nepřerušil další pokusy; doba zpracování prvního a posledního požadavku byla srovnatelná. V tomto testovaném vzorku tedy nebyla pozorována žádná ochrana proti automatizovanému zkoušení hesel.

Praktický dopad

Pokud se toto chování potvrdí i mimo testovaný vzorek, může útočník proti známým e-mailům ve velkém zkoušet uniklá a slabá hesla (credential stuffing). Riziko roste, pokud uživatelé recyklují hesla z jiných služeb.

Důkaz a reprodukce

Předpoklady: jeden platný e-mail

Krok 1: odeslat 200 přihlašovacích požadavků s různými hesly v rychlém sledu (sekvenčně, bez pauzy)

Výsledek: všech 200 požadavků vráceno s kódem 401, žádný s kódem 429; doba prvního a posledního požadavku byla srovnatelná

Zachycený průběh:

```
# 200 pokusů o přihlášení v rychlém sledu; sledujeme kód i dobu odpovědi
$ for i in $(seq 1 200); do
    curl -s -o /dev/null -w "%{http_code} %{time_total}\n" \
      -X POST https://app.scheduly.example/api/auth/login \
      -H 'Content-Type: application/json' \
      -d "{\"email\":\"user-a@scheduly.example\",\"password\":\"x$i\"}"
  done | awk '{n[$1]++; if(NR==1)f=$2; l=$2}
              END{for(k in n) print n[k], k;
                  printf "prvni=%ss posledni=%ss\n", f, l}'
200 401
prvni=0.049s posledni=0.048s

# Pozorování z tohoto konkrétního vzorku (200 sekvenčních pokusů):
# žádný požadavek nevrátil 429, doba odezvy mezi prvním a posledním pokusem
# se nelišila. Test necílil na zjištění cool-off po vyšším počtu pokusů
# ani na přítomnost captcha – o tom tento vzorek nic nedokládá.
```

Doporučení k nápravě

Zavést adaptivní throttling podle více signálů (IP, zařízení, chování), rostoucí prodlevu a captcha při opakovaném selhání. Vyhnout se tvrdému zamykání účtu jen podle e-mailu, aby nešlo cíleně blokovat cizí účty (account-lockout DoS). Zvážit upozornění uživatele na neobvyklé pokusy. Očekávaný stav: automatizované zkoušení hesel je zpomaleno nebo blokováno, aniž by šlo snadno zablokovat cizí účet.

Ověření opravy

Sada rychlých chybných pokusů narazí na limit nebo blokaci.

F-04 Zjistitelnost existence účtů (user enumeration)

Pole	Obsah
Závažnost	STŘEDNÍ
Stav	OTEVŘENO
Dotčená část	POST /api/auth/login, POST /api/auth/reset
Kategorie	Autentizace / únik informací
OWASP / CWE	A07:2025 Authentication Failures · CWE-204
Důvěra v nález	Potvrzeno

Popis

Přihlášení a reset hesla vracejí odlišné odpovědi podle toho, zda účet existuje („neznámý e-mail“ vs. „nesprávné heslo“). Podle hlášky lze ověřit, které e-maily jsou v systému registrované.

Praktický dopad

Umožňuje sestavit seznam platných účtů pro cílený phishing a pro útok z nálezu F-03. Sám o sobě nevede k úniku dat, ale zesiluje ostatní útoky.

Důkaz a reprodukce

Předpoklady: jeden existující a jeden neexistující e-mail

Krok 1: na přihlášení i resetu odeslat neexistující e-mail → odpověď „neznámý účet“

Krok 2: odeslat existující e-mail → odlišná odpověď („špatné heslo“ / „e-mail odeslán“)

Výsledek: rozdíl v odpovědi a HTTP kódu prozrazuje existenci účtu

Zachycený průběh:

```
# Přihlášení s NEexistujícím e-mailem
$ curl -s -w " [%{http_code}]\n" -X POST \
       https://app.scheduly.example/api/auth/login \
       -H 'Content-Type: application/json' \
       -d '{"email":"nikdo@scheduly.example","password":"x"}'
{"error":"unknown_email"} [404]

# Přihlášení s existujícím e-mailem a špatným heslem
$ curl -s -w " [%{http_code}]\n" -X POST \
       https://app.scheduly.example/api/auth/login \
       -H 'Content-Type: application/json' \
       -d '{"email":"user-a@scheduly.example","password":"x"}'
{"error":"invalid_password"} [401]

# Reset hesla pro NEexistující účet
$ curl -s -w " [%{http_code}]\n" -X POST \
       https://app.scheduly.example/api/auth/reset \
       -H 'Content-Type: application/json' \
       -d '{"email":"nikdo@scheduly.example"}'
{"error":"account_not_found"} [404]

# Reset hesla pro EXISTUJÍCÍ účet
$ curl -s -w " [%{http_code}]\n" -X POST \
       https://app.scheduly.example/api/auth/reset \
       -H 'Content-Type: application/json' \
       -d '{"email":"user-a@scheduly.example"}'
{"status":"email_sent"} [200]

# Přihlášení i reset prozrazují (odpovědí a HTTP kódem), které účty existují.
```

Doporučení k nápravě

Sjednotit odpovědi bez ohledu na existenci účtu (u resetu vždy „pokud účet existuje, poslali jsme e-mail“; u přihlášení vždy generická chyba) a sjednotit i časování odpovědí. Očekávaný stav: z odpovědi nelze rozlišit existující a neexistující účet.

Ověření opravy

Odpovědi i jejich časování jsou pro existující i neexistující účty shodné.

F-05 Detailní chybové odpovědi

Pole	Obsah
Závažnost	NÍZKÁ
Stav	OTEVŘENO
Dotčená část	/api/* (chybové stavy)
Kategorie	Konfigurace / únik informací
OWASP / CWE	A02:2025 Security Misconfiguration · CWE-209
Důvěra v nález	Potvrzeno

Popis

Při chybě aplikace vrací podrobný výpis (stack trace) s interními cestami a hlavičku X-Powered-By s názvem frameworku. Chování bylo pozorováno na testovaném (staging) prostředí.

Praktický dopad

Podrobné chyby prozrazují použitý framework, interní cesty a strukturu aplikace, což útočníkovi usnadňuje mapování technologie a cílení jejích známých slabin. Pokud stejná konfigurace běží i v produkci, je tento informační únik dostupný komukoli, kdo vyvolá chybu. Nejde o přímý průnik, ale snižuje laťku pro další útoky.

Důkaz a reprodukce

Krok 1: odeslat neplatný vstup na endpoint (např. nesprávný datový typ v těle)

Výsledek: odpověď obsahuje stack trace, interní cesty a hlavičku s názvem frameworku

Zachycený průběh:

```
# Neplatný datový typ v těle požadavku
$ curl -i -s -X POST https://app.scheduly.example/api/invoices \
       -H 'Content-Type: application/json' \
       -d '{"amount":"not-a-number"}'
HTTP/1.1 500 Internal Server Error
X-Powered-By: Express
Content-Type: text/html; charset=utf-8

TypeError: expected number, received string
    at InvoiceService.create (/srv/app/services/invoice.js:88:14)
    at handler (/srv/app/routes/invoices.js:23:9)
    ...

# Odpověď vrací stack trace a interní cesty; hlavička X-Powered-By prozrazuje framework.
```

Doporučení k nápravě

Vypnout detailní chybové výpisy mimo vývojové prostředí, vracet generickou zprávu a chybu logovat pouze na serveru. Ověřit, že produkční konfigurace detailní chyby nezobrazuje. Očekávaný stav: klient dostává generickou chybu bez interních detailů.

Ověření opravy

Vyvolané chyby v testovaném prostředí nevracejí stack trace ani hlavičku prozrazující framework; produkční konfiguraci ověřit samostatně (vyžaduje přístup k produkci).

7. Pozitivní zjištění

Přístup k datovým endpointům vyžaduje přihlášení; slabinou je až chybějící kontrola vlastnictví (F-01), ne samotná autentizace.

HTTPS je vynuceno na celé aplikaci; přesměrování z HTTP funguje.

Session cookie má nastavené atributy HttpOnly a Secure.

8. Doporučený plán nápravy

Priorita	Doporučený krok	Vlastník	Cílový stav
1	Ověřovat vlastnictví záznamů serverově na všech /api/* endpointech (F-01)	Vývoj	Cizí ID vrací 403/404 bez dat
1	Role řídit serverově; neodvozovat je z dat od klienta (F-02)	Vývoj	Uživatel si nemůže změnit vlastní roli
2	Zavést rate limiting a prodlevu na přihlášení (F-03)	Vývoj / Ops	Zkoušení hesel je zpomaleno nebo blokováno
3	Sjednotit chybové hlášky a časování odpovědí (F-04)	Vývoj	Nelze zjistit existenci účtu
3	Vypnout detailní chybové výpisy mimo vývojové prostředí (F-05)	Ops	Klient dostává generickou chybu

9. Závěr

U obou kritických nálezů (F-01 a F-02) se opakuje společný problém: na dotčených endpointech není dostatečně vynucováno oprávnění na straně serveru — aplikace se spoléhá na to, že klient požadavek mimo svá práva neodešle. Doporučuji stejný vzor prověřit i na dalších endpointech nad rámec této kontroly. Zbývající nálezy (F-03 až F-05) se týkají odlišných oblastí — ochrany autentizace proti automatizovaným pokusům, zjistitelnosti účtů a rozsahu chybových odpovědí — a je třeba je řešit samostatně. Doporučené pořadí oprav: nejdříve F-01 a F-02, poté F-03 a F-04, nakonec F-05. Po opravě obou kritických nálezů doporučuji cílený retest dotčených toků. Rozsah odpovídá službě Bezpečnostní kontrola webové aplikace; hlubší prověření nad rámec tohoto rozsahu je možné řešit samostatně jako Penetrační test na míru.

10. Přílohy

Klasifikace závažnosti

Závažnost je kombinací praktického dopadu a snadnosti zneužití.

Závažnost	Kritérium
KRITICKÁ	Přímý únik dat nebo převzetí účtu či aplikace, snadno zneužitelné bez zvláštních podmínek.
VYSOKÁ	Závažný dopad na data nebo oprávnění, zneužitelné s malým úsilím.
STŘEDNÍ	Reálný dopad za určitých podmínek nebo v kombinaci s dalším nálezem.
NÍZKÁ	Omezený dopad; spíše zvyšuje riziko, než působí přímou škodu.
INFORMAČNÍ	Bez přímého rizika; doporučení k dobré praxi.

Použité nástroje

- Burp Suite — zachytávání a úprava HTTP požadavků
- curl — ruční reprodukce požadavků
- ffuf — kontrolované procházení endpointů a ID
- Prohlížeč s vývojářskými nástroji

Nástroje slouží jako podpora ruční práce, ne jako zdroj automaticky generovaných nálezů. Každý nález byl ověřen ručně.

Testované endpointy (výběr)

```
POST   /api/auth/login
POST   /api/auth/reset
GET    /api/invoices/{id}
POST   /api/invoices
PATCH  /api/users/me
GET    /api/admin/users
GET    /api/clients
web:   /login  /invoices  /clients  /settings
```

Testovací účty

Bez hesel: user-a@scheduly.example, user-b@scheduly.example, admin-demo@scheduly.example

Informace o dodavateli

Dodavatel
SAM SPACE e.d. s.r.o.
IČ: 03593975
Sídlo: Mládeže 1342/23, 169 00 Praha 6
Nejsem plátce DPH

Kontaktní údaje
Milan Půlkrábek
milan@digitalnisebeobrana.cz
+420 776 636 110
digitalnisebeobrana.cz