Biztonsági szkenner

Alap Windows védelmi ellenőrző alkalmazás fejlesztése (SecurityShell)

**Írta:**

Németh Attila

**Neptun kód:**

M1PCDI

Tartalomjegyzék

[Tartalomjegyzék ii](#_Toc199039994)

[1. Bevezetés 1](#_Toc199039995)

[1.1 Az informatikai rendszerek védelmének fontosága 1](#_Toc199039996)

[1.2 A Windows operációs rendszerek biztonsági beállításainak ellenőrzése 1](#_Toc199039997)

[1.3 A célalkalmazás bemutatása: SecurityShell 1](#_Toc199039998)

[1.4 A dolgozat célja és felépítése 1](#_Toc199039999)

[2. Elméleti háttér 3](#_Toc199040000)

[2.1 A Windows biztonsági modellje 3](#_Toc199040001)

[2.2 Windows Defender 3](#_Toc199040002)

[2.3 Windows Update 4](#_Toc199040003)

[2.4 Windows Tűzfal 4](#_Toc199040004)

[2.5 Felhasználói hitelesítés és bejelentkezési védelem 4](#_Toc199040005)

[3. Gyakorlati megvalósítás 5](#_Toc199040006)

[3.1 Szabályok – HA/AKKOR típusú logika és jogosultság 6](#_Toc199040007)

[3.2 Az alkalmazás működésének folyamata (folyamatábra) 7](#_Toc199040008)

[3.3 Windows Defender állapotának ellenőrzése 9](#_Toc199040009)

[3.4 Felhasználói felület 9](#_Toc199040010)

[4. Tesztelés 10](#_Toc199040011)

[5. Eredmények és kiértékelés 11](#_Toc199040012)

[6. Következtetés és jövőbeli fejlesztési lehetőségek 12](#_Toc199040013)

[6.1 Fejlesztési irányok 12](#_Toc199040014)

[6.2 HA-AKKOR szabálybázis és intelligencia 12](#_Toc199040015)

[6.3 Piacképesség és bővíthetőség 12](#_Toc199040016)

[7. Források 13](#_Toc199040017)

# 1. Bevezetés

## 1.1 Az informatikai rendszerek védelmének fontosága

A digitális világ rohamos fejlődése, a felhasználói adatok tömeges keletkezése, valamint az internetre kapcsolt eszközök számának robbanásszerű növekedése napjainkra jelentős kihívások elé állította az informatikai rendszerek biztonságát. Az otthoni és vállalati számítógépek egyaránt ki vannak téve különböző típusú fenyegetéseknek, mint például vírusoknak, kémprogramoknak, zsarolóvírusoknak (ransomware), valamint hálózati támadásoknak. Ennek következtében a számítógépek védelmének alapját képező rendszerbeállítások ellenőrzése és karbantartása kulcsfontosságú feladattá vált.

## 1.2 A Windows operációs rendszerek biztonsági beállításainak ellenőrzése

A Microsoft Windows operációs rendszere – különösen a 10-es és 11-es verziók – egy sor beépített biztonsági funkciót kínálnak, mint például a Windows Defender, a tűzfal, a felhasználói fiókok védelme (UAC), illetve az automatikus frissítési mechanizmusok. Bár ezek a szolgáltatások megfelelően konfigurálva képesek alapvető védelmet nyújtani, a felhasználók többsége nem rendelkezik elegendő szakmai ismerettel ahhoz, hogy megbizonyosodjon ezen beállítások helyes működéséről vagy hogy felismerje az esetleges hiányosságokat.

## 1.3 A célalkalmazás bemutatása: SecurityShell

A „SecurityShell” nevű alkalmazás célja, hogy egyszerű, grafikus felhasználói felületen keresztül átfogó képet nyújtson a Windows alapvető biztonsági beállításainak állapotáról, és amennyiben problémát észlel, a felhasználót érthető módon tájékoztassa, illetve – ha szükséges – segítséget is nyújtson a hiba elhárításához. Az alkalmazás fejlesztése során elsődleges szempont volt a felhasználóbarát kialakítás, az informatikában kevésbé jártas felhasználók támogatása, valamint a Windows rendszer natív szolgáltatásainak elérhetősége és lekérdezhetősége a .NET keretrendszer és PowerShell parancsok segítségével.

## 1.4 A dolgozat célja és felépítése

A dolgozat célja, hogy bemutassa a SecurityShell alkalmazás létrejöttének hátterét, a tervezési és megvalósítási folyamatokat, valamint részletesen ismertesse, milyen módszerekkel és technológiákkal ellenőrizhetők a Windows rendszer legfontosabb biztonsági funkciói. A szoftver célcsoportja elsősorban otthoni felhasználók, rendszergazdák és informatikus hallgatók, akik szeretnének átfogó képet kapni számítógépük védelmi állapotáról anélkül, hogy bonyolult rendszerkonfigurációkba kellene beavatkozniuk.

A dolgozat további fejezetei részletesen kitérnek az egyes vizsgált védelmi elemekre – mint például a valós idejű vírusvédelem, automatikus frissítések, tűzfal állapot, valamint a bejelentkezési védelem. Ezen felül szó esik a felhasználói élmény szempontjairól, a fejlesztési kihívásokról, valamint a jövőbeli fejlesztési lehetőségekről is.

A SecurityShell egy olyan projekt, amely egyszerre kíván technikai és felhasználói oldalról is választ adni a modern számítógépes védelem alapkérdéseire, és demonstrálja, hogyan lehet egy informatikai alkalmazás fejlesztésével kézzelfogható megoldást nyújtani egy aktuális problémára.

# 2. Elméleti háttér

A számítógépes rendszerek biztonsága a digitális korszakban elengedhetetlen, különösen az otthoni és vállalati környezetekben, ahol érzékeny adatok, munkaállományok, valamint személyes és üzleti információk kerülnek tárolásra és feldolgozásra. A Windows operációs rendszerek – amelyek a világ egyik legelterjedtebb platformjai – beépített és opcionális védelmi mechanizmusokat kínálnak a felhasználók biztonságának megőrzésére. A „SecurityShell” elnevezésű program célja ezen védelmi funkciók meglétének és állapotának egyszerű, áttekinthető ellenőrzése.

## 2.1 A Windows biztonsági modellje

A Microsoft Windows egy többrétegű biztonsági modellt alkalmaz, amely különböző komponensekből áll össze. Az operációs rendszer feladata, hogy biztosítsa a rendszer integritását, megvédje a felhasználói adatokat, és lehetőséget nyújtson az illetéktelen hozzáférések elleni védelemre. A biztonsági modell elemei:

* Felhasználói jogosultságok kezelése: Az operációs rendszer támogatja a felhasználói fiókok típusait (pl. normál, rendszergazda), valamint a jogosultságok pontos konfigurálását.
* Hitelesítés és engedélyezés: A bejelentkezési mechanizmus és jelszókezelés alapvető fontosságú a felhasználói identitás ellenőrzéséhez.
* Hozzáférés-szabályozási lista (ACL): A fájlokhoz, mappákhoz és rendszerobjektumokhoz való hozzáférést ACL szabályozza.
* Folyamatos rendszerfrissítés: A Windows Update rendszeres biztonsági javításokat biztosít, melyek célja a szoftverhibák és sérülékenységek megszüntetése.

## 2.2 Windows Defender

A Windows Defender (teljes nevén: Microsoft Defender Antivirus) a Microsoft által fejlesztett beépített vírusirtó és kártevő-elhárító megoldás. A Windows 10 óta teljes értékű védelmi eszközként működik, amely valós idejű védelmet biztosít. Képes automatikus keresésekre, karanténba helyezésre, és veszélyforrás-elemzésre. A Defender három fő funkcionális területet fed le:

* Valós idejű védelem: Folyamatos monitorozás az ismert fenyegetések ellen.
* Időzített ellenőrzések: Előre beállított időpontokban történő rendszerellenőrzés.
* Fenyegetéskezelés: Karantén, kártevők eltávolítása, jelentések készítése.

Amennyiben egy másik vírusirtó szoftver van telepítve, a Defender automatikusan inaktívvá válik, hogy ne ütközzön más védelmi megoldással.

## 2.3 Windows Update

A Windows Update szolgáltatás feladata, hogy az operációs rendszerhez és a Microsoft szoftverekhez szükséges frissítéseket – ideértve a biztonsági javításokat, hibajavításokat és funkciófrissítéseket – eljuttassa a felhasználókhoz. Az automatikus frissítések bekapcsolt állapota kiemelten fontos a sebezhetőségek gyors befoltozása érdekében.

A Windows Update típusai:

* Kritikus frissítések – azonnali telepítést igénylő biztonsági javítások.
* Minőségi frissítések – a rendszer stabilitását és teljesítményét javítják.
* Funkciófrissítések – az operációs rendszer új verziói, jellemzően évente jelennek meg.

A felhasználók dönthetnek az automatikus, halasztott vagy kézi frissítés mellett, azonban a biztonság szempontjából az automatikus opció javasolt.

## 2.4 Windows Tűzfal

A Windows tűzfal (Windows Defender Firewall) célja, hogy szabályozza a hálózati kommunikációt az eszköz és a külvilág között. Beállításaitól függően képes blokkolni a nem megbízható alkalmazások adatforgalmát, ezzel megelőzve a távoli támadásokat.

A tűzfal három hálózati profil alapján működik:

* Privát hálózat
* Nyilvános hálózat
* Tartományi hálózat

Minden profilhoz külön szabálykészlet tartozik. A felhasználónak lehetősége van engedélyezni vagy tiltani programokat, portokat, vagy IP-cím tartományokat. A tűzfal állapotának helyes működése alapvető fontosságú a hálózati támadások megelőzése szempontjából.

## 2.5 Felhasználói hitelesítés és bejelentkezési védelem

A felhasználói hitelesítés célja annak biztosítása, hogy a rendszerhez kizárólag jogosult személyek férhessenek hozzá. A Windows lehetőséget nyújt jelszavas, PIN-kódos, biometrikus (pl. ujjlenyomat, arcfelismerés) és kétlépcsős hitelesítési megoldásokra. Ezek védelmet nyújtanak a fizikai hozzáférés elleni támadásokkal szemben is.

A leggyakoribb támadási forma a „brute-force” jelszótörési technika, mely ellen az erős jelszavak, illetve a fiókzárolási irányelvek (pl. 5 sikertelen próbálkozás után fiókzárolás) jelentenek megoldást.

# 3. Gyakorlati megvalósítás

A SecurityShell nevű alkalmazás megalkotásának célja egy olyan egyszerű, de hatékony biztonsági szkenner létrehozása, amely képes a Windows operációs rendszer alapvető védelmi beállításainak állapotát gyorsan és felhasználóbarát módon ellenőrizni. Az alkalmazás C# nyelven, Windows Forms (WinForms) technológiával készült, amely megfelelő eszközkészletet biztosít a grafikus felhasználói felület megvalósításához, miközben könnyen karbantartható és továbbfejleszthető szerkezetet nyújt.

**A gyakorlati megvalósítás során a projekt legfontosabb szempontjai a következők voltak:**

* Átláthatóság: A felhasználó egyértelmű visszajelzést kapjon a védelmi állapotokról.
* Egyszerű használat: Néhány kattintással végrehajthatók legyenek az ellenőrzések.
* Segítségnyújtás: Amennyiben egy adott védelmi komponens inaktív, az alkalmazás ne csak jelezze a problémát, hanem nyújtson útmutatást is annak megoldásához.
* Kiterjeszthetőség: A kód olyan struktúrában készült, hogy további funkciók is könnyen hozzáadhatók legyenek (például víruskereső szoftverek felismerése, rendszerleíró kulcsok ellenőrzése stb.).

**A fejlesztés során az alkalmazás négy fő ellenőrzési pontot valósít meg:**

* Windows Defender állapota: A rendszer alapértelmezett valós idejű védelmi moduljának aktív státuszát ellenőrzi PowerShell parancs segítségével.
* Automatikus frissítések állapota: Lekérdezi, hogy a Windows Update szolgáltatás megfelelően be van-e állítva és működik-e.
* Windows tűzfal állapota: A rendszerbe épített tűzfal aktív státuszának vizsgálata történik.
* Felhasználói fiók védettség: A jelszavas bejelentkezés és a felhasználói fiókok biztonsági szintjének ellenőrzése.

Ezen vizsgálatok megvalósításához a Windows által biztosított API-k, PowerShell parancsok, valamint WMI (Windows Management Instrumentation) lekérdezések kerültek felhasználásra. A különböző komponensek eredményét az alkalmazás vizuálisan is megjeleníti (pl. színkódolás, szöveges visszajelzés), és ha szükséges, lehetőséget biztosít a részletes segítség megjelenítésére is.

A következőkben részletesen bemutatásra kerül mindegyik vizsgálat külön alfejezetben, a mögöttes technikai megvalósítással és a használt kódrészletekkel együtt.

## 3.1 Szabályok – HA/AKKOR típusú logika és jogosultság

Példaszabály (Windows Defender):

HA a Windows Defender valós idejű védelme inaktív,

AKKOR a rendszer védelem nélkül marad, így a felhasználót értesíteni kell, és javasolni a védelem bekapcsolását.

A Microsoft hivatalos dokumentációja szerint a Defender valós idejű védelmének aktívnak kell lennie minden időpontban, hogy a kártevők elleni védelem folyamatosan biztosított legyen.

## 3.2 Az alkalmazás működésének folyamata (folyamatábra)

**Folyamatábra**



1. A SecurityShell alkalmazás működési folyamatábrája

**Alkalmazás működési folyamata**

Az alábbi ábra szemlélteti a SecurityShell nevű biztonsági ellenőrző alkalmazás működésének alapfolyamatát. A program felhasználóbarát, grafikus felületen teszi lehetővé az egyes védelmi komponensek manuális ellenőrzését.

A program elindítása után a felhasználó a gomb segítségével hajthat végre célzott biztonsági vizsgálatokat:

* Windows Defender ellenőrzés – megvizsgálja, hogy a Microsoft beépített valós idejű vírusvédelme aktív állapotban van-e. Az eredmény a felületen szöveges formában jelenik meg.
* Antivírus szoftverek listázása – kiírja, hogy milyen egyéb vírusvédelmi szoftverek vannak telepítve a rendszeren.
* Windows Update ellenőrzés – lekérdezi az automatikus frissítések beállításait, és figyelmeztetést ad, ha a frissítési szolgáltatás nem működik megfelelően.

Minden ellenőrzés eredménye külön mezőben jelenik meg (label1–label3), és a felhasználónak egyértelmű visszajelzést ad az adott funkció állapotáról.

Ez a megközelítés különösen alkalmas olyan felhasználók számára, akik egyszerű, lépésenkénti ellenőrzést szeretnének végezni a rendszerük alapvető biztonsági beállításairól.

**Interaktív visszajelzés és súgó**

A SecurityShell alkalmazás nem csupán ellenőrzi a rendszer biztonsági beállításait, hanem valós idejű visszajelzést is nyújt a felhasználónak az eredményekről. Amennyiben valamelyik védelmi komponens – például a Windows Defender – inaktív vagy hibás állapotban van, a program egy felugró ablakban részletes útmutatást jelenít meg.

Ez a „súgóablak” tartalmazza a szükséges lépéseket az adott probléma megoldásához, például a valós idejű védelem újbóli aktiválásához. A súgó célja, hogy segítséget nyújtson a nem szakértő felhasználóknak, akik nem feltétlenül ismerik a Windows beállítási lehetőségeit.

Az alábbi ábra egy példát mutat a Windows Defender inaktív állapota esetén megjelenő súgóablakról:



2. A Windows Defender inaktív állapota esetén megjelenő súgóablak

A felhasználó a súgóban szereplő „Windows Defender megnyitása” gombra kattintva közvetlenül elérheti a vonatkozó rendszerbeállításokat, így nem szükséges manuálisan keresgélnie az operációs rendszer beállításai között. Ez a funkció jelentősen javítja a felhasználói élményt és elősegíti a problémák gyors megoldását.

A súgó egy interaktív megoldási javaslatot kínáló funkció.

**Mögötte is egy szabály áll:**

HA a Defender inaktív,

AKKOR jelenjen meg egy figyelmeztető ablak a problémával és egy „Beállítások megnyitása” gombbal.

Ez a „HA–AKKOR” szabály logikailag beépített reakció a programban, amely nemcsak a hiba tényét közli, hanem egyúttal konkrét útmutatást is nyújt a javításhoz.

## 3.3 Windows Defender állapotának ellenőrzése

A SecurityShell egyik elsődleges feladata annak ellenőrzése, hogy ez a védelem aktív állapotban van-e, és biztosítja-e a rendszer folyamatos felügyeletét.

A Windows Defender állapotának lekérdezéséhez PowerShell parancsot használ az alkalmazás, amelyet a .NET keretrendszer System.Management.Automation nevű könyvtárán keresztül hív meg. Az alábbi PowerShell parancs segítségével lekérdezhető, hogy a Defender be van-e kapcsolva és aktív-e a valós idejű védelem:



A DisableRealtimeMonitoring tulajdonság False értéke esetén a Defender aktív, míg True esetén a valós idejű védelem ki van kapcsolva. Ez kulcsfontosságú információ a biztonság szempontjából, hiszen ennek hiányában a rendszer nyitottá válik a kártevők számára.

## 3.4 Felhasználói felület

A felhasználói felület egyszerűen kezelhető bárki számára. A Defender állapotát egy szöveges doboz mondja el, valamint egy „Súgó” felületet is biztosít, amely részletes útmutatást nyújt a felhasználónak, hogyan tudja manuálisan bekapcsolni a védelmet, ha az kikapcsolt.

Az alkalmazás egy extra gombot is tartalmaz, amely lehetővé teszi, hogy a felhasználó egyetlen kattintással megnyissa a Windows Defender biztonsági beállításait. Ez a következő paranccsal érhető el:



# 4. Tesztelés

# 5. Eredmények és kiértékelés

# 6. Következtetés és jövőbeli fejlesztési lehetőségek

A SecurityShell jelenlegi verziója manuális, gombalapú biztonsági ellenőrzéseket tesz lehetővé. Minden vizsgálat eredménye külön mezőben jelenik meg, így a felhasználó egyértelmű szöveges visszajelzést kap a rendszerállapotról.

## 6.1 Fejlesztési irányok

A közeljövőben számos fejlesztési lehetőség kínálkozik a program bővítésére. Ilyen lehet például:

* automatikus összesítő kiértékelés beépítése, amely átfogó képet adna a rendszer biztonsági állapotáról,
* színkódolt visszajelzés a felhasználó számára (pl. zöld–sárga–piros státusz),
* automatikus beavatkozási lehetőségek, például a Windows Defender bekapcsolása inaktív állapot esetén.

## 6.2 HA-AKKOR szabálybázis és intelligencia

A program logikai működése tovább bővíthető lenne egy szabálybázis segítségével, amely HA–AKKOR típusú feltételek alapján képes különböző rendszerállapotokra reagálni. Például:

HA egy rendszerállapot vagy szolgáltatás inaktív,

AKKOR a program figyelmeztesse a felhasználót, és tegye lehetővé a beállítások elérését az adott komponensnél.

Ezek a szabályok nemcsak reagálni tudnak a hibás állapotra, hanem önállóan is képesek lehetnek javító lépések végrehajtására.

## 6.3 Piacképesség és bővíthetőség

A SecurityShell hosszabb távon piacképes segédalkalmazássá fejleszthető, különösen, ha:

* a szabályalapú működés kiegészül adatbázisban tárolt biztonsági eseményekkel,
* a program pontozási rendszert vezet be, amely százalékos formában mutatja a gép védettségi szintjét,
* és automatikusan be is tud avatkozni bizonyos hibák esetén (pl. szolgáltatás engedélyezése, beállítás megnyitása).

A program jövőbeli célja egy olyan intelligens, önálló döntéshozatalra képes alkalmazás kialakítása, amely nemcsak figyelmeztet, hanem segít megoldani a problémát – felhasználói beavatkozás nélkül vagy azzal együtt.

# 7. Források

1. **Microsoft Learn – Windows biztonsági architektúra**

URL: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-security-architecture

1. **Microsoft Docs – Microsoft Defender Antivirus Overview**

URL: https://learn.microsoft.com/en-us/microsoft-365/security/defender-endpoint/microsoft-defender-antivirus

1. **Microsoft Support – Windows Update: FAQ**

URL: https://support.microsoft.com/en-us/windows/windows-update-faq

1. **Microsoft Learn – Windows Defender Firewall**

URL: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-firewall/windows-firewall-with-advanced-security

1. **Microsoft Learn – Windows Hello és felhasználói hitelesítés**

URL: https://learn.microsoft.com/en-us/windows/security/identity-protection/hello-for-business/hello-overview

1. **National Institute of Standards and Technology (NIST) – Password Guidelines**

URL: https://pages.nist.gov/800-63-3/

1. **ChatGPT, Copilot – További szöveges források**