1 Threat Analysis

1.1 Authentication

According to the Annual Cyber Threat Report(ACTR) 2023-2024, the most frequently reported threat in Australia were Authentication attacks in the form of "compromised accounts or credentials"[1], accounting for nearly one-third of all cybercrime incidents nationwide. This prevalence underscores the critical importance of understanding and mitigating authentication-related threats in the current cybersecurity landscape.

Authentication breaches encompass a wide array of attack categories, each exploiting different weaknesses in user and system behavior. The most prominent include **phishing**, which leverages psychological manipulation to trick users into divulging credentials, and **brute force attacks**, where adversaries systematically guess passwords using automated tools. **Password spraying** represents a "low and slow" approach, targeting many accounts with common passwords to evade detection. **Credential stuffing** utilizes previously breached username-password pairs, often sourced from large-scale data leaks, to gain unauthorized access through automation. These categories highlight the multifaceted nature of authentication threats, demonstrating why password-based attacks remain a critical focus for security research and organizational defense.

1.2 Threats in the Authentication Landscape

1.2.1 Current Threat Landscape

Authentication threats represent a critical challenge in today's cybersecurity landscape, both in Australia and globally. The Albanese Government's commitment of $15–$20 billion through 2033–34 to strengthen cyber domain capabilities underscores the national urgency to address these issues[1]. In the 2023–24 financial year, the Australian Signals Directorate received over 36,700 calls to its Cyber Security Hotline 12% increase from the previous year, reflecting the growing prevalence of cyber incidents[1]. Notably, compromised accounts or credentials accounted for **32% of all cybercrime**, making it the leading contributor to reported incidents. On a global scale, Microsoft Entra data reveals that of more than **600 million identity attacks per day**, over **99% are password-based**[2]. These statistics highlight the widespread and persistent nature of authentication-related threats, demonstrating why robust authentication mechanisms are essential for organizational and national security.

1.2.2 Typical Adversary Types

Authentication attacks are executed through several distinct methods, each exploiting specific vulnerabilities in user behavior and system design.

- **Phishing** is a psychological manipulation technique where adversaries impersonate trusted entities, often via email or messaging platforms, to deceive users into revealing their credentials. This method remains highly effective due to its reliance on social engineering rather than technical flaws.
- **Brute force attacks** involve systematically generating and attempting password combinations until a valid login is achieved. Attackers leverage automated tools to accelerate this process, targeting accounts with weak or commonly used passwords.
- **Password spraying** refines brute force tactics by distributing login attempts across many accounts using a shortlist of popular passwords, thereby evading detection mechanisms that monitor for rapid, repeated failures. A notable example is the 2024 Midnight Blizzard incident, where attackers employed password spraying against Microsoft's infrastructure, prompting significant defensive actions[3].
- **Credential stuffing** utilizes credentials harvested from previous data breaches, exploiting the widespread practice of password reuse across multiple services. The 2022 Optus breach illustrated the impact of this technique, as compromised credentials were repurposed to target other organizations[4]. These methods collectively demonstrate the evolving sophistication and persistence of authentication threats in the modern cyber landscape.

1.2.3 Impact

This section starts with how phishing and credential stuffing can be used as good starts

The organizational impact of authentication breaches is starkly illustrated by the 2022 Optus incident, which exposed sensitive data of nearly 10 million Australians and triggered widespread concerns over identity theft and fraud [4]. More recently, the Qantas breach in July 2025 affected 5.7 million customers, with varying degrees of personal information compromised [5]. Both cases highlight the persistent threat posed by credential stuffing, where attackers leverage previously leaked credentials to infiltrate systems at scale. The consequences include

substantial financial losses, operational disruption, and enduring reputational harm, emphasizing the critical need for organizations to strengthen authentication controls and proactively address credential-based threats.

1.3 Threat Choice Justification

The selection of authentication attacks as the focal threat for this analysis is driven by their overwhelming prevalence and critical impact on both Australian and global organizations. Recent reports indicate that compromised credentials account for nearly one-third of all cybercrime incidents in Australia, with high-profile breaches such as Optus and Qantas underscoring the real-world consequences of these attacks[1], [4], [5]. The universal reliance on password-based authentication across industries makes this threat highly relevant and widely applicable. From a technical perspective, authentication attacks such as brute force, password spraying, and credential stuffing offer a rich landscape for analysis, enabling clear success and failure metrics and supporting laboratory-based experimentation.

While phishing remains a significant aspect of authentication security due to its effectiveness and prevalence, it is not easily testable in a controlled lab environment and will not be the focus of practical testing in this assignment. Nevertheless, its role in the broader authentication threat landscape will be acknowledged. Furthermore, the threat aligns strongly with established frameworks like MITRE ATT&CK (T1110 family)[6], providing a robust foundation for evaluating defensive strategies and mapping adversary techniques.

This combination of prevalence, technical depth, and framework integration makes authentication attacks an ideal subject for comprehensive security analysis. It will also be interesting to see how easy it is to break into hypothetical systems and gain a practical understanding of how secure my own personal passwords really are.

2 Tool Comparison

2.1 Overview of Attacker Tools

2.2 Comparison Table

Tool	**Attack Type**	**Primary Use Case**	**Key Strengths**
Hashcat	Offline hash cracking	Password recovery from hashes/encrypted files	450+ algorithms, GPU acceleration, high performance
Hydra	Online network attacks	Live service authentication testing	50+ protocols, real-time feedback, flexible attack modes
John the Ripper	Offline hash cracking	Educational/legacy password auditing	Rule engine, single mode, broad compatibility
NetExec	Network reconnaissance	Enterprise Active Directory assessment	Multi-protocol, BloodHound integration, lateral movement

Table 1: Attack Tools Comparison

2.3 Hashcat

Hashcat is a specialized tool for password-based attacks, focusing on scenarios where the attacker has access to known hash or encrypted data and aims to recover the original password or passphrase. It employs brute force, dictionary, and hybrid attack methods to achieve this goal. Hashcat excels at cracking password hashes across more than 450 supported algorithms[7], recovering passwords for encrypted file formats, and deciphering offline authentication tokens.

Hashcat is limited to offline password recovery and cannot break mathematically sound encryption without access to passwords, attack live network services, or exploit flaws in cryptographic implementations. Its strengths lie in unmatched password recovery capabilities, broad algorithm support, and GPU-accelerated performance, making it a specialized tool for cracking password hashes and encrypted files rather than attacking network protocols or cryptographic weaknesses.

Rather than replacing network-based tools such as Hydra. Hashcat focuses on offline attacks against captured password hashes or encrypted files, while Hydra targets live authentication services by attempting to guess

credentials in real time. Hashcat operates independently of network access, working in the dark to recover passwords from stored data. Once successful, the recovered credentials can be used to access systems without further brute force attempts.

2.3.1 Strengths and Weaknesses

Strengths:

- Unmatched password recovery capabilities
- Supports 450+ password hash algorithms
- GPU-accelerated performance for rapid attacks
- Effective for encrypted file formats and offline authentication tokens

Limitations:

- Requires access to password-protected material (hashes, encrypted files)
- Cannot break mathematically sound encryption without passwords
- Not suitable for attacking live network services
- Does not exploit cryptographic implementation flaws

2.3.2 Use Cases

- Cracking password hashes extracted from sources such as Windows "Ntds.dit" or "SAM" files, Linux "/etc/shadow", or other stored hash values supported by Hashcat's extensive algorithm coverage.
- Recovering lost passwords for encrypted files or authentication tokens

2.4 Hydra

Hydra is a versatile tool for conducting password-based attacks against live network services. Unlike Hashcat, which focuses on offline password recovery from hashes or encrypted files, Hydra targets active authentication endpoints by systematically testing username and password combinations in real time. It supports more than 50 protocols, including web services (HTTP, HTTPS), remote access (SSH, Telnet, RDP), file sharing (FTP, SMB), databases (MySQL, PostgreSQL), and email (IMAP, SMTP)[8]. Hydra is commonly used for brute force, password spraying, and credential stuffing attacks, providing immediate feedback on successful logins.

For this research, Hydra was chosen over Medusa[9] due to Medusa's lack of recent git commits and lower comparative popularity on GitHub, indicating less active development and community support. Medusa offers functionality comparable to Hydra, with a modular architecture that enables consistent command usage across various protocols. However, its lower popularity and less active development made it a secondary choice for this research.

2.4.1 Strengths and Weaknesses

Strengths:

- Directly tests live authentication systems
- Supports a wide range of network protocols
- No need for hash extraction
- Immediate identification of valid credentials
- Flexible attack modes (brute force, spraying, credential stuffing)
- Custom wordlist and username/password combinations

Limitations:

- Generates network traffic that can be detected and logged
- May trigger account lockout or security alerts
- Network speed and server response time can affect performance
- Cannot bypass multi-factor authentication or modern protocols (OAuth, SAML)
- Requires network connectivity to the target

2.4.2 Use Cases

- Testing the strength of passwords for web applications, remote access, and file sharing services
- Simulating password spraying attacks to evaluate organizational defenses
- Assessing exposure to credential stuffing

2.5 John the Ripper

John the Ripper is a foundational password cracking and security auditing tool that established many baseline techniques in the field. As the "original" password cracker[10], it operates primarily through offline hash cracking similar to Hashcat, while also offering limited online attack capabilities. John the Ripper exists in two main variants: the classic mode with traditional password cracking algorithms, and the community-enhanced "Jumbo" version that supports over 200 hash formats including Unix/Linux systems, Windows environments, applications, databases, and cryptocurrency wallets[11].

John the Ripper distinguishes itself through its sophisticated rule engine for password transformations and unique "single mode" that leverages account information to generate targeted password candidates. While it shares offline hash cracking capabilities with Hashcat, John the Ripper's historical significance and educational value make it valuable for understanding classical password cracking techniques, though it operates at significantly slower speeds on modern GPU hardware.

2.5.1 Strengths and Weaknesses

Strengths:

- Historical significance with established algorithms and techniques
- Sophisticated rule engine for password transformation
- Unique single mode using account information for targeted attacks
- Broad compatibility with older and limited hardware
- Educational value with well-documented classical algorithms
- Dual architecture supporting both classic and enhanced modes

Limitations:

- Significantly slower GPU performance compared to Hashcat
- Slower development pace and update cycle
- More complex configuration than modern tools
- Less optimized memory efficiency for large-scale attacks
- Limited online attack capabilities compared to dedicated network tools

2.5.2 Use Cases

- Educational environments for learning password cracking fundamentals
- Legacy system auditing where modern tools may not be compatible
- Specialized rule-based attacks leveraging account information
- Cryptocurrency wallet password recovery for older formats
- Cross-platform password auditing on resource-constrained systems

2.6 NetExec

NetExec is a network penetration testing tool designed to identify security vulnerabilities within enterprise networks by systematically testing credentials across multiple services and protocols. Operating under the philosophy that "you're only as strong as your weakest point," NetExec performs comprehensive credential validation across network infrastructure to locate authentication weaknesses[12]. The tool primarily targets Windows-specific protocols commonly used in enterprise environments, making it particularly effective for Active Directory assessments and post-exploitation activities.

NetExec distinguishes itself from single-protocol tools like Hydra through its multi-protocol approach and specialized Windows/Active Directory focus. While Hydra targets individual services, NetExec provides a comprehensive network-wide assessment capability, automatically testing credentials against discovered services and integrating with tools like BloodHound for attack path visualization[13]. This makes it valuable for both initial network reconnaissance and post-compromise lateral movement scenarios.

2.6.1 Strengths and Weaknesses

Strengths:

- Supports multiple protocols (SMB, LDAP, WINRM, MSSQL, SSH, FTP, RDP, WMI, NFS)
- Integrated BloodHound support for network mapping and attack path visualization
- Automated credential validation across entire network infrastructure
- Specialized Windows/Active Directory enumeration capabilities
- Post-exploitation lateral movement through advanced credential attacks

- Support for pass-the-hash, Kerberos abuse, and LAPS integration

Limitations:

- Primarily Windows/Active Directory focused, limiting applicability to Linux-only environments
- Requires existing network access for post-exploitation activities
- May trigger security alerts due to multiple authentication attempts
- Cannot bypass multi-factor authentication or modern OAuth/SAML protocols
- Dependent on network connectivity and target system availability

2.6.2 Use Cases

- Enterprise Active Directory assessment for automating credential validation and security posture evaluation across large Windows networks
- Post-compromise lateral movement through advanced credential attacks including pass-the-hash and Kerberos abuse
- Comprehensive domain security validation including delegation detection and Certificate Services enumeration
- Automated BloodHound data collection for attack path visualization and privilege escalation planning

2.7 Attack Tool Selection and Summary

After evaluating the four authentication attack tools, **Hydra** emerged as the most suitable choice for this research assignment. This selection was critical to establish before proceeding to defensive tool analysis, as it provides the foundation for understanding what threats need to be mitigated.

Hydra offers the ideal balance of capabilities for this assignment's scope. Its support for over 50 network protocols, real-time authentication testing, and immediate feedback mechanisms make it particularly valuable for understanding live network attack scenarios. The tool's ability to perform brute force, password spraying, and credential stuffing attacks directly aligns with the authentication threat landscape identified in the threat analysis section.

NetExec presented fascinating capabilities, particularly its comprehensive Active Directory assessment features and BloodHound integration. However, its scope proved too extensive for this assignment's constraints, encompassing post-exploitation lateral movement and enterprise-wide network reconnaissance that extends beyond focused authentication testing. While an extremely powerful tool for real-world penetration testing, its complexity would have diluted the research focus.

Hashcat demonstrated impressive technical capabilities with its 450+ algorithm support and GPU acceleration. Despite being a foundational tool in password recovery, it lacked the network-based attack vectors I sought to explore in this assignment. Its offline nature, while valuable for hash cracking scenarios, doesn't provide the interactive network authentication testing central to this research.

John the Ripper, while historically significant as the original password cracker, unfortunately appeared outdated compared to modern alternatives like Hashcat. Though it retains educational value and offers unique features like its sophisticated rule engine, its slower performance and less active development made it a secondary choice for contemporary security research.

This tool selection process ensured that the subsequent defensive analysis would address the most relevant and practical authentication threats

3 Authentication Attack Scenario Design - Dot Point Plan

3.1 Testing Scenario Overview

Objective: Evaluate effectiveness of password spraying attacks using personalized password generation against various defensive measures

3.2 Environment Setup

3.2.1 Virtual Lab Infrastructure

- **Target System:** Ubuntu server with SSH and web application login
- **Attack Platform:** Kali Linux VM with custom tools
- **Network:** Host-only networking (isolated environment)
- **Monitoring:** Centralized logging server for detection analysis

3.2.2 Target Applications

- **Primary:** Custom web application with login form
- **Secondary:** SSH service with username/password authentication
- **Database:** MySQL backend storing user credentials (hashed)

3.3 Phase 1: Synthetic Data Generation

3.3.1 User Profile Database Creation

- Generate **500 realistic user profiles** using Python Faker library
- **Data points per user:**
  - Full name (first, last)
  - Email address
  - Birth year/date
  - Pet names (1-2 per user)
  - Hobbies/interests
  - Favorite foods
  - Location/city
  - Company/job title
  - Previously gained password

3.3.2 Password Generation Algorithm

- **Pattern-based password creation** using personal data
- **Common password patterns to implement:**
  - `{FirstName}{BirthYear}!` (e.g., Sarah1990!)
  - `{Pet}{Number}@` (e.g., Fluffy123@)
  - `{Hobby}{Year}` (e.g., Photography2024)
  - `{Location}{BirthYear}!` (e.g., Sydney1985!)
  - `{Company}@{Year}` (e.g., Microsoft@2024)
  - `{PreviousPassword}` (If available, used once, attributes can also be dissected and rearranged, (e.g., Candy*2000 could be dissected into `[Candy,$,2000]`))
- **Ensure compliance with password policy:** minimum 8 chars, uppercase, lowercase, number, special character

3.4 Phase 2: Attack Implementation (Offensive Tool)

3.4.1 Custom Password Spraying Tool

- **Language:** Python with requests/paramiko libraries
- **Features:**
  - Multi-protocol support (HTTP/SSH)
  - Configurable timing delays (avoid detection)
  - User-agent rotation and IP spoofing simulation
  - Success/failure logging with timestamps
  - Pattern effectiveness tracking

3.4.2 Attack Methodology

- **Stage 1:** Username enumeration (if required)
- **Stage 2:** Single password against all users (classic spraying)
- **Stage 3:** Personalized passwords against specific users
- **Timing:** 1 attempt per user per 30 minutes (avoid lockouts)
- **Duration:** 48-hour attack simulation

3.5 Phase 3: Defensive Implementation

3.5.1 Baseline Testing (No Defenses)

- Document attack success rate without protection
- Establish baseline metrics for comparison

3.5.2 Defense Tool 1: fail2ban

- **Configuration:**
  - SSH protection: 3 failures = 10 minute ban
  - HTTP protection: 5 failures = 15 minute ban
  - Custom rules for web application
- **Monitoring:** Ban events, IP addresses, timing

3.5.3 Defense Tool 2: Account Lockout Policies

- **Application-level:** 5 failed attempts = 30 minute lockout
- **System-level:** PAM modules for SSH protection
- **Progressive delays:** Increasing wait times per failed attempt

3.5.4 Defense Tool 3: Enhanced Monitoring

- **Log analysis:** Automated pattern detection
- **Alerting:** Real-time notifications for spray patterns
- **SIEM simulation:** Correlation rules for attack identification

3.6 Success Metrics Definition

3.6.1 Attack Success Metrics (Offensive)

- **Primary:** Successful authentication rate (logins/total attempts)
- **Secondary:** Time to first successful login
- **Pattern effectiveness:** Which personal data patterns yield highest success
- **Evasion rate:** Attacks completed without triggering defenses

3.6.2 Defense Success Metrics (Defensive)

- **Primary:** Attack detection rate (alerts/actual attacks)
- **Secondary:** Time to detection (minutes from attack start)
- **False positive rate:** Legitimate users blocked
- **Containment effectiveness:** Attacks stopped vs completed

3.6.3 Overall Effectiveness Metrics

- **Business impact:** Percentage of accounts compromised
- **Operational impact:** Legitimate user disruption
- **Detection accuracy:** True positives vs false positives

3.7 MITRE ATT&CK Framework Mapping

3.7.1 Primary Techniques

- **T1110.003 - Brute Force: Password Spraying**
  - Main attack vector implementation
- **T1589.001 - Gather Victim Identity Information: Credentials**
  - Personal data collection and analysis
- **T1078.004 - Valid Accounts: Cloud Accounts**
  - Successful authentication usage

3.7.2 Supporting Techniques

- **T1201 - Password Policy Discovery**
  - Understanding target password requirements
- **T1087 - Account Discovery**
  - Username enumeration activities
- **T1133 - External Remote Services**
  - SSH/web service targeting

3.8 Testing Timeline

3.8.1 Week 1: Environment setup and data generation

3.8.2 Week 2: Attack tool development and testing

3.8.3 Week 3: Baseline attack execution (no defenses)

3.8.4 Week 4: Defense implementation and protected testing

3.8.5 Week 5: Analysis, comparison, and documentation

3.9 Expected Outcomes

3.9.1 Attack Tool Analysis

- Personalized passwords significantly more effective than generic wordlists
- Lower detection rates with slow, targeted approach
- Pattern effectiveness varies by demographic correlation

3.9.2 Defense Tool Analysis

- fail2ban effective against rapid attacks, less effective against slow spraying
- Account lockouts protect individual accounts but may cause DoS
- Combined defenses provide layered protection with acceptable false positive rates

3.9.3 Scenario Winner Determination

- **Attacker wins:** >10% successful authentication rate
- **Defender wins:** <2% successful authentication rate + >90% detection rate
- **Balanced outcome:** 2-10% success rate with high detection

```
import flask
```

References

[1] Australian Cyber Security Centre, "Annual cyber threat report 2023-2024," Australian Signals Directorate, 2024. Available: https://www.cyber.gov.au/about-us/view-all-content/reports-and-statistics/annual-cyber-threat-report-2023-2024

[2] Microsoft Corporation, "Microsoft digital defense report 2024," Microsoft Corporation, Technical Report, 2024. Available: https://www.microsoft.com/en-us/security/security-insider/threat-landscape/microsoft-digital-defense-report-2024

[3] Microsoft Security Response Center, "Microsoft actions following attack by nation state actor midnight blizzard." 2024. Available: https://msrc.microsoft.com/blog/2024/01/microsoft-actions-following-attack-by-nation-state-actor-midnight-blizzard/

[4] D. Maguire, "What's happening with the optus data breach? What we know about the alleged hacker's ransom, data release and apology," *ABC News*, 2022, Available: https://www.abc.net.au/news/2022-09-27/optus-data-breach-cyber-attack-hacker-ransom-sorry/101476316

[5] S. Chalmers, "Qantas confirms 5.7 million customers were impacted in cyber attack," *ABC News*, 2025, Available: https://www.abc.net.au/news/2025-07-09/qantas-confirms-number-of-customers-impacted-in-cyber-attack/105510654

[6] MITRE Corporation, "MITRE ATT&CK technique T1110: Brute force." 2024. Available: https://attack.mitre.org/techniques/T1110/

[7] J. Steube, "Hashcat: Advanced password recovery." 2025. Available: https://hashcat.net/hashcat/

[8] van Hauser, "THC-hydra: Online password cracking tool." 2025. Available: https://github.com/vanhauser-thc/thc-hydra

[9] F. Networks, "Medusa: Parallel network login auditor." 2025. Available: http://foofus.net/?page_id=51

[10] S. Designer, "John the ripper: Password cracker." 2025. Available: https://github.com/openwall/john

[11] KeychainX, "How to recover lost bitcoin passwords." 2021. Available: https://keychainx.medium.com/how-to-recover-lost-bitcoin-passwords-c34c42ee6f17

[12] NetExec Team, "NetExec wiki." 2024. Available: https://www.netexec.wiki/

[13] SpecterOps, "BloodHound legacy." 2024. Available: https://github.com/SpecterOps/BloodHound-Legacy