Pentest & Red-Team Toolkit Automation — Summary

Author: Velkris (Jared Perry)
Focus: Offensive Security Engineering / Environment Automation
Environment: Custom Kali Linux toolkit with Bash, PowerShell, and Ansible provisioning
Purpose: Streamline setup, configuration, and operation of a standardized red-team toolkit for repeatable lab and assessment workflows.

Objective

Design and implement a modular automation framework to deploy a consistent, mission-ready red-team toolkit across multiple lab environments—reducing manual setup time and configuration drift.

Toolkit Architecture

Component	Function
Base OS	Kali Linux (rolling) with custom hardening and persistent storage
Automation Layer	Bash and PowerShell scripts orchestrated through Ansible
Tool Categories	Reconnaissance, exploitation, post-exploitation, reporting
Version Control	Git-based synchronization and tool version pinning

---

Methodology

- Automated provisioning of reconnaissance, exploitation, and reporting tools via scripted installers.
- Integrated common frameworks (Impacket, CrackMapExec, Evil-WinRM, Burp Suite, BloodHound).
- Implemented configuration checks and log validation for consistency.
- Documented SOPs to support repeatable deployments across virtualized environments.

Tools & Frameworks

Ansible • Bash • PowerShell • Git • Python • Kali Linux • BloodHound • CrackMapExec • Burp Suite • Evil-WinRM

Results

Metric	Outcome
Deployment Time	Reduced by ~50% through scripted provisioning.
Consistency	Identical toolset and configuration across all test VMs.
Reusability	Version-controlled scripts enable easy rebuilds for future labs.

Recommendations

- Maintain version-locked manifests for tool dependencies.
- Expand automation to include reporting templates and evidence archiving.
- Integrate with CI/CD pipelines for long-term red-team infrastructure readiness.

Outcome

Delivered a **standardized, automated toolkit** improving readiness for red-team engagements and certification labs.
Enabled consistent recon-to-report workflows and simplified maintenance of multiple offensive environments.

> This project highlights technical efficiency, reproducibility, and disciplined engineering within offensive-security operations.