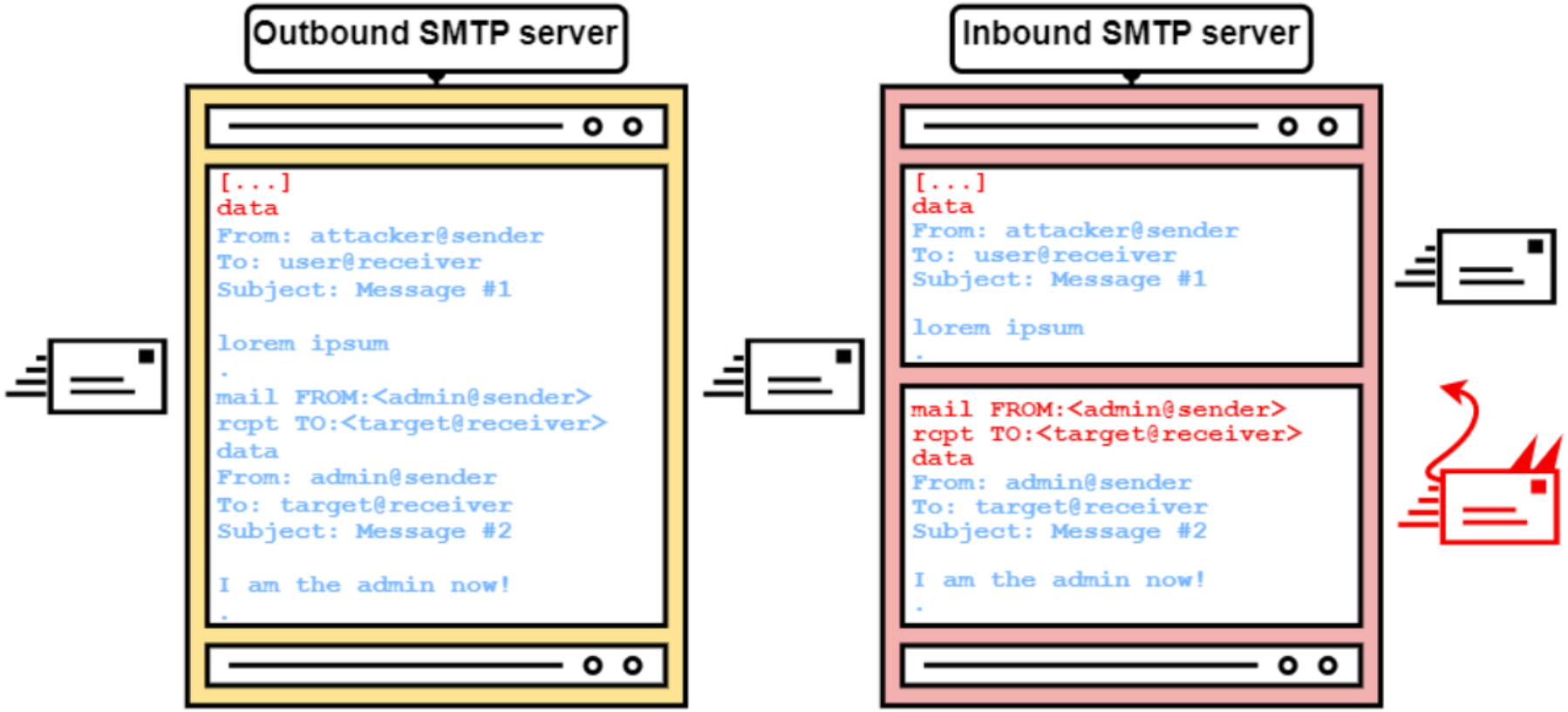
Email Smuggling with Differential Fuzzing of MIME Parsers

Seyed Behnam Andarzian, Martin Meyers, Erik Poll

IEEE Symposium on Security and Privacy Workshops - LangSec 2025

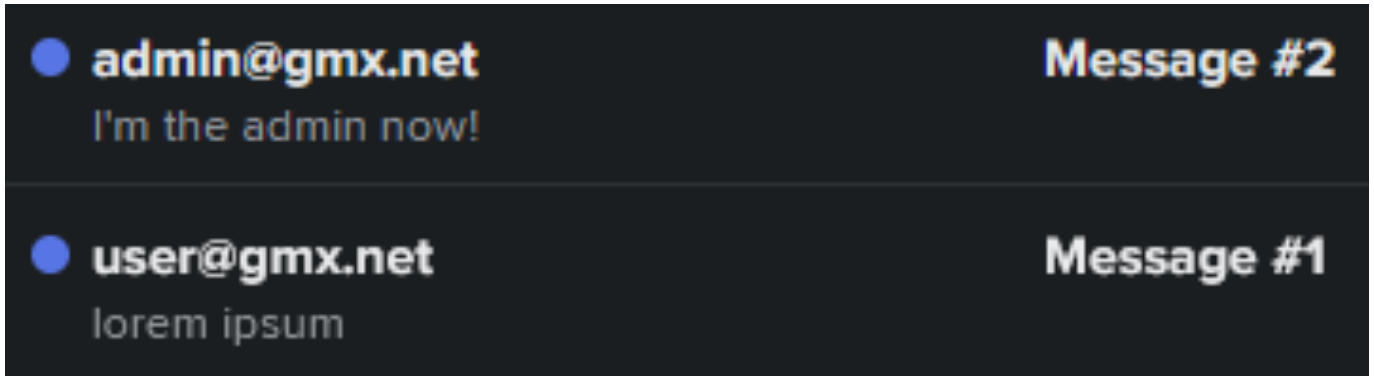
Radboud University

Email Smuggling:

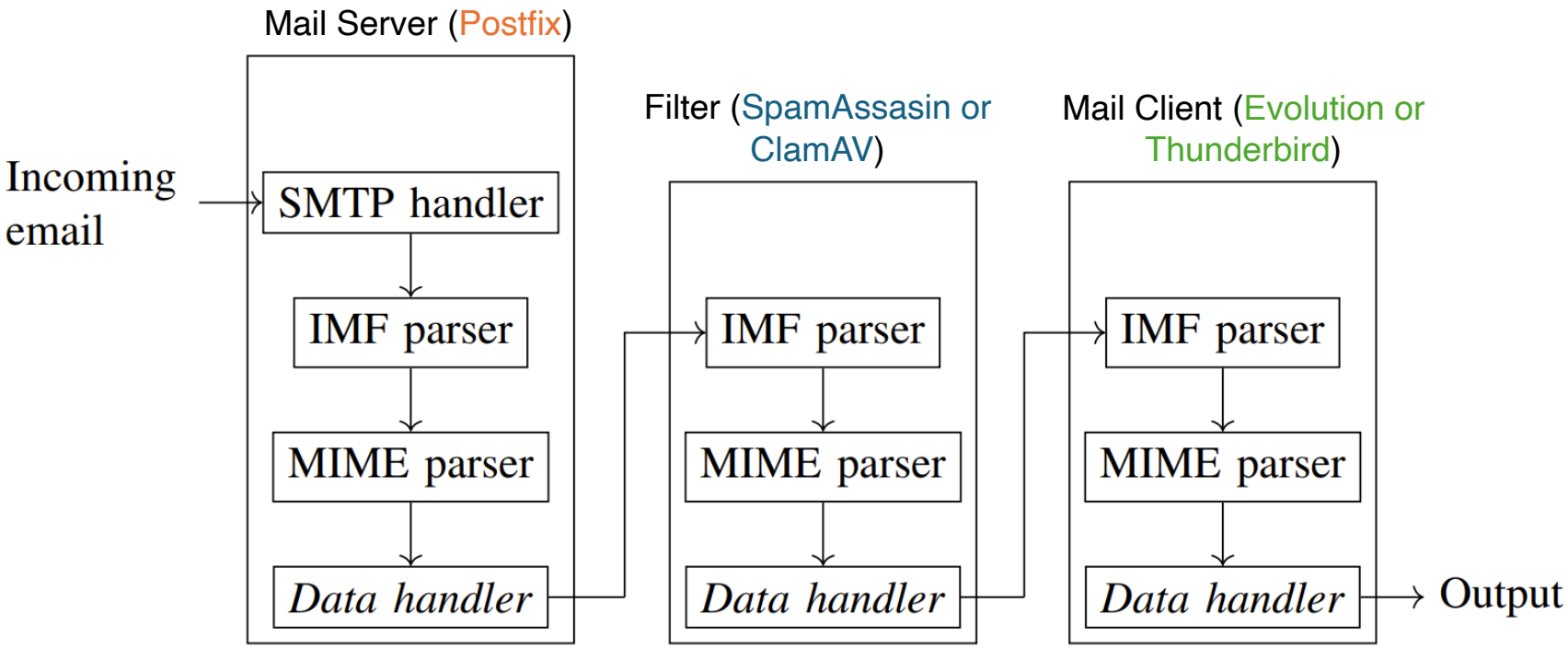


Email Smuggling:

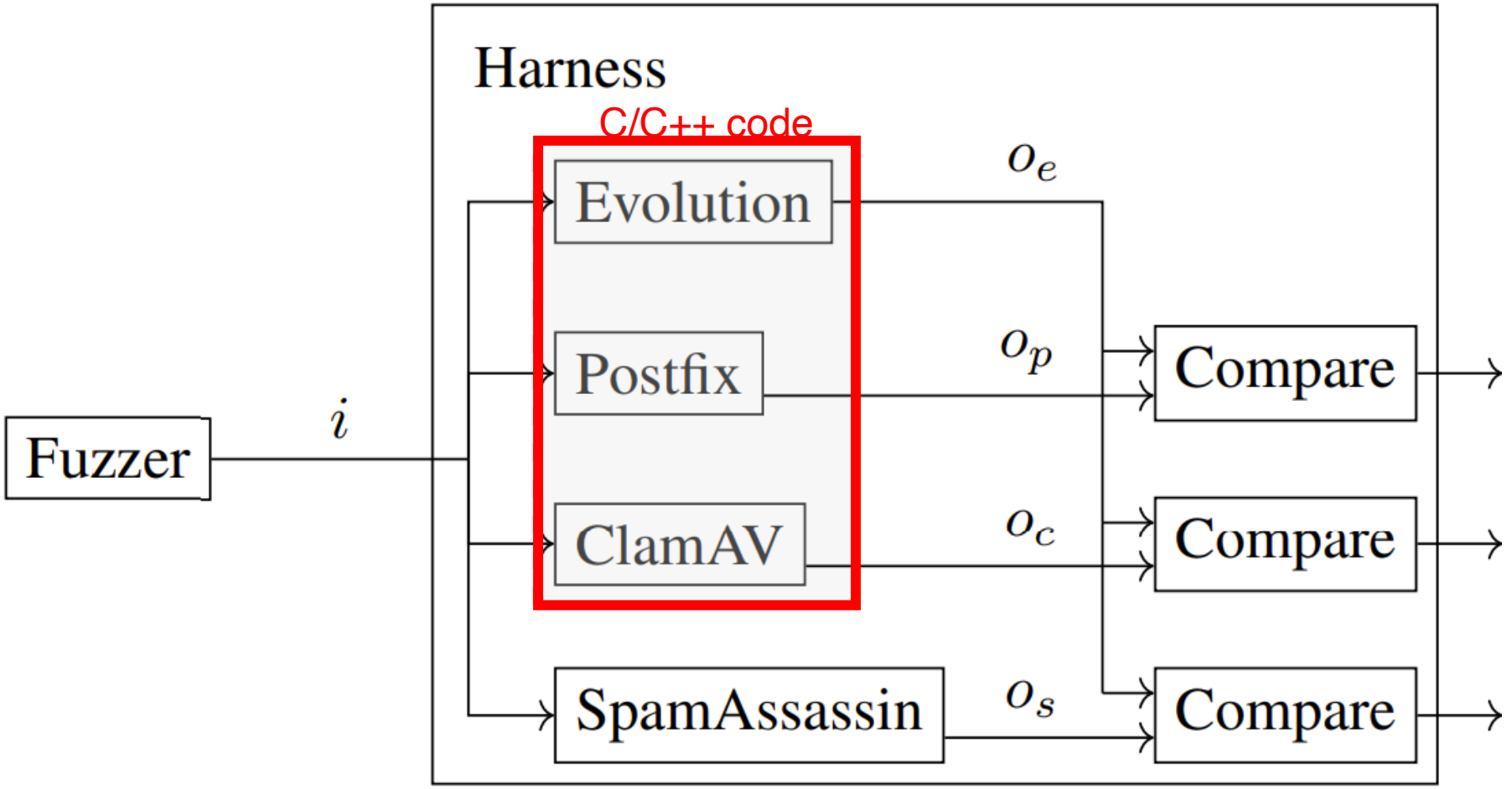
The user sees two emails!



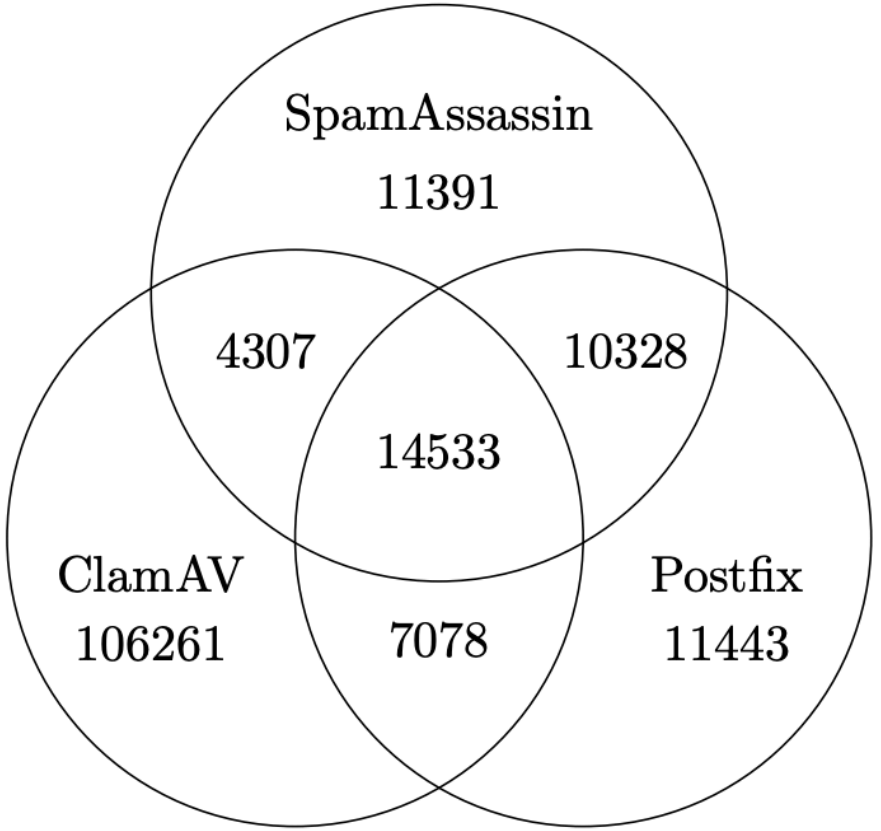
Email Handling:



Differential Fuzzing:

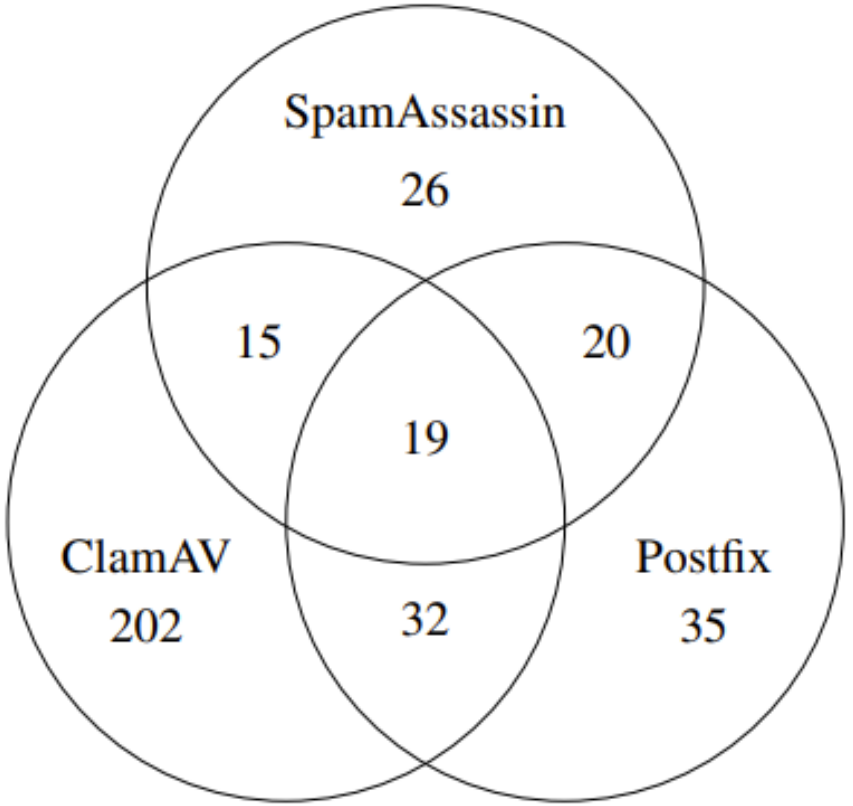


Differentials found with T-Reqs before de-duplication:



- Evolution is the ground truth

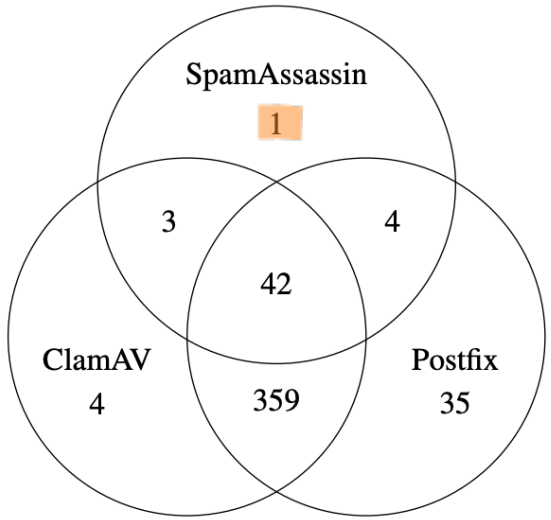
Differentials found with T-Reqs after de-duplication with afl-cmin:



- Evolution is the ground truth

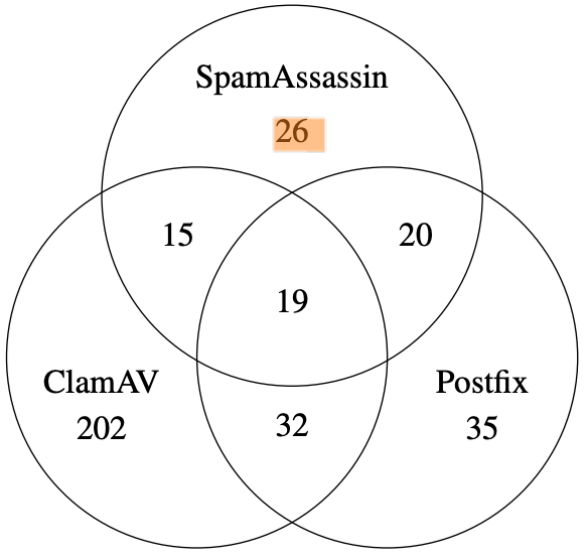
Grey-box fuzzer: AFL++

- More differentials for C/C++targets



Grammar based fuzzer: T-Reqs

- Differentials are more meaningful
  == more likely to be exploitable



Root causes of the differentials :

1. Malformed headers
2. Different decoding strategies

Root cause	Postfix	SpamAssassin	ClamAV	Thunderbird
D1	✓			
D2	✓			
D3		✓		
D4		✓	✓	
D5		✓	✓	
D6		✓		
D7	✓		✓	✓
D8	✓	✓		
D9			✓	
D10			✓	
D11		✓		✓
D12	✓	✓	✓	✓
D13			✓	
D14		✓		✓
D15	✓	✓	✓	✓

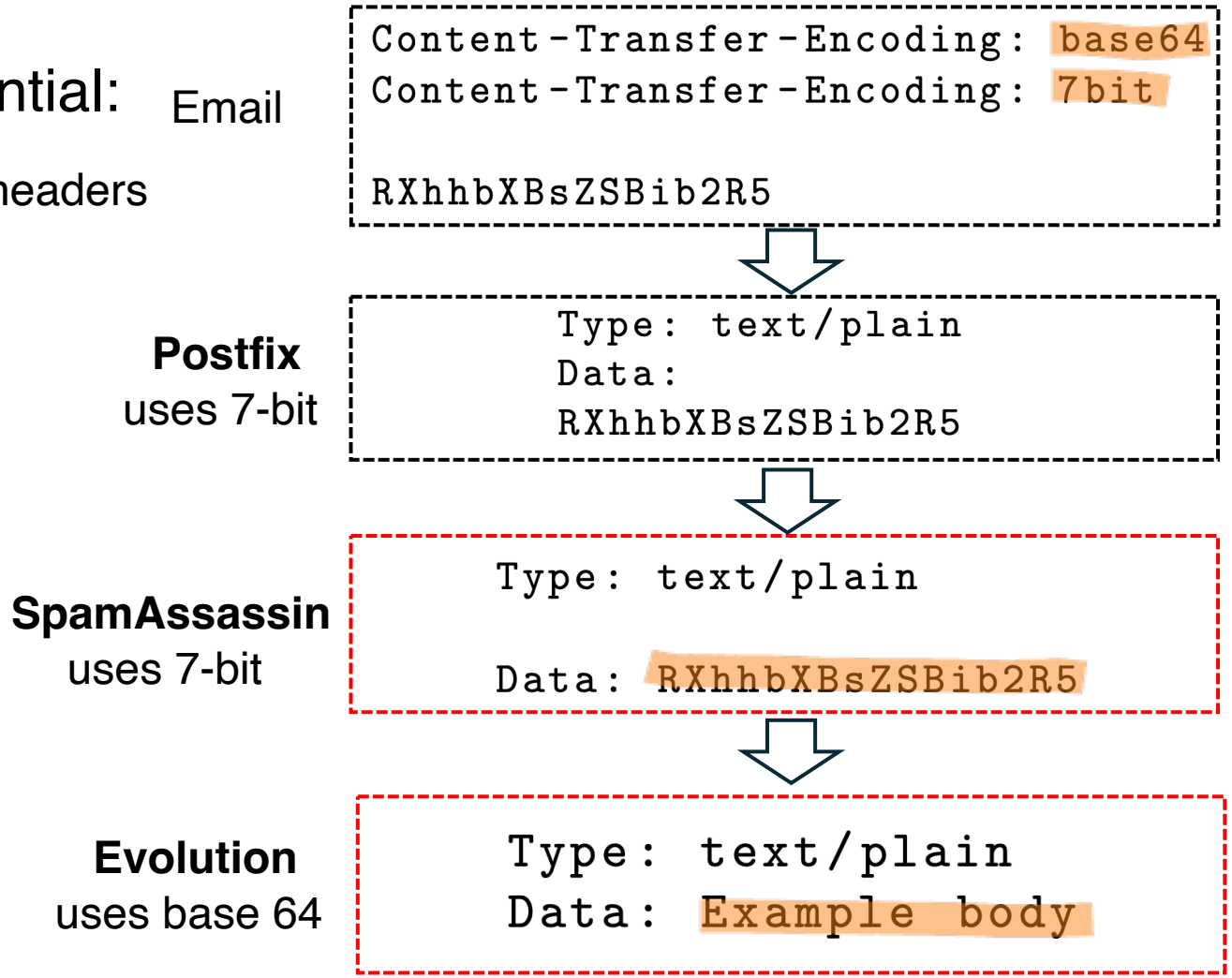
Exploitability results:

Difference cause	Evolution	Thunderbird
D1		
D2		
D3	✓	✓
D4	✓	✓
D5		
D6		
D7		
D8		
D9		
D10		
D11	✓	
D12		
D13		
D14		
D15		

(a) Smuggling through SpamAssassin

Example exploitable differential:

Multiple content transfer encoding headers



Memory corruptions bugs found by AFL++:

- Two memory-corruption vulnerabilities found in Evolution and ClamAV
- Numerous assertion failures found in both Evolution and ClamAV
- ClamAV is enrolled in OSS-fuzz project, but these vulnerabilities are not found
    - The test harness is poor?

Key takeaways:

- MIME specs are sloppy
- Lots of differentials ☹
- Few exploitable ☺
- Lot of work to look into the differentials
- Grammar based fuzzer is better in finding exploitable differentials
- Surprisingly, we found memory corruption bugs; don't these people use fuzzers?!

Thank you