Smart's Attack Timeline - Real-time Execution Flow

Server Start

Server initializes vulnerable ECC curve
Trace of Frobenius = 1

Info Request

Client requests public key
and curve parameters

Vulnerability Check

Client verifies:
trace of Frobenius = 1

Scale Points

Compute p×P' and p×Q'
in p-adic field

Verification

Verify: P × n = Q
✓ Success!

Decrypt

Client decrypts using
recovered private key

Client Connect

Client establishes connection
to server

Info Response

Server sends curve params:
p, a, b, P, Q

Lift to p-adic

Lift curve and points
to $\mathbb{Q}_p$ field

Discrete Log

Calculate private key:
$n = (Q'_x/Q'_y)/(P'_x/P'_y)$

Challenge

Server sends encrypted
challenge message

Success

FLAG revealed!
Attack completed

t0 t1 t2 t3 t4 t5 t6 t7 t8 t9 t10 t11

SETUP **ATTACK** **EXPLOIT**