Formal code security

Challenge: code vulnerabilities
The FormalLab team was tasked to examine the use of code analysis tools for cybersecurity purposes.
We confronted security components and their requirements to the Frama-C source code analysis platform.

Innovation: mathematical guarantees
Plain requirements are lifted into precise, unambiguous formal specifications. The launch of dedicated analyzers provides mathematical guarantees that the software matches its specifications.

```c
//@ requires @ (md->curlen < sizeof(md->buf));
error_t SHA256_update(SHA256_ctx_t *md, const uint8_t *data, ..);
//@ assert mem_access: \valid(x[i+1]);
x[i+1] |= x[i] >> 31;
//@ assigns *Qx \From pECOMInstance, *pECOMInstance, d, *d,
d_bytelen;
error_t ECOM_keyDerivation_ByteArray(EC_Domain_t *pECOMInstance,
uint8_t *d, unsigned long d_bytelen, uint8_t *Qx, uint8_t *Qy);
```

Perspectives
Thales develops further pilots to validate operational usages of formal methods in development processes.
The FormalLab team assists Thales in this mission, and prepares to tackle the next generation of challenges in software and systems verification.

Contact
florent.kirchner@cea.fr
jean-marc.mota@thalesgroup.com