TraceComp Unified Master Document

Founder: Fuci Yamani Sofyan

Date Started: May 2025

# 1. Executive Summary

TraceComp is a cybersecurity framework designed to embed attribution metadata into compiled binaries at the build stage.  
It shifts malware analysis from reactive static/dynamic inspection to proactive authorship visibility, enabling SOC teams,  
Red Teams, and researchers to instantly trace binary origin, intent, and identity. Every malware will now have a ‘digital birth certificate’.

# 2. Sejarah & Tujuan Projek

TraceComp bermula daripada satu persoalan kritikal: Kenapa selepas malware menyerang, baru kita sibuk nak static dan dynamic analysis?  
Kenapa tidak dari awal lagi kita tahu siapa compile, bila dan dari mana? Dari situ lahir idea bahawa setiap malware seperti manusia perlu ada  
‘surat beranak’. TraceComp direka untuk menyuntik metadata identiti semasa proses kompilasi, supaya fail binari membawa tag asal-usul yang boleh  
dikesan, divisualisasikan dan diverifikasi.

# 3. Infrastruktur & Reka Bentuk Projek

Struktur projek:  
- builder/ : tracecomp\_builder.py (suntik metadata)  
- extractor/ : tracecomp\_extractor.py (baca metadata)  
- samples/: contoh binary  
- plugins/: sokongan untuk compiler seperti PyInstaller, Go, GCC  
- docs/: dokumentasi penuh dan spesifikasi teknikal  
- tests/: ujian unit dan simulasi  
- README.md dan LICENSE

# 4. Struktur Metadata

Contoh:  
{  
 "ORG": "RedOps",  
 "COMPILED\_BY": "yamani\_fuci",  
 "CAMPAIGN": "MAY25-RTOC",  
 "BUILD\_TIME": "2025-05-20T13:00Z",  
 "TRACE\_ID": "TC-24052025-01"  
}

# 5. Branding: TraceComp vs GhostTag

TraceComp adalah nama rasmi projek dalam konteks akademik dan pembangunan terbuka. GhostTag adalah branding stealth khusus untuk Red Team tools,  
di mana metadata boleh di-obfuscate untuk digunakan dalam simulated adversarial ops tanpa kompromi identiti dalaman.

# 6. Security Model & Mitigasi Risiko

- Tamper Awareness: Metadata mengandungi checksum atau hash untuk kesan ubahsuai.  
- Signature Layer: Metadata boleh disahkan dengan kunci peribadi untuk pastikan kesahihan.  
- Metadata Spoof: Jika metadata diubah, hasil verification akan gagal.  
- Clone Protection: Tanpa kunci sah, sistem SOC boleh kesan metadata tiruan melalui mismatch struktur logik atau validasi kriptografi.

# 7. AI Integration & Automation

Projek ini menggunakan AI untuk mempercepatkan pembangunan, dokumentasi, penjanaan metadata dan analisis.   
Contoh aplikasi AI:  
- Auto-generate metadata tags berdasarkan fail hash atau nama user  
- Tulis dokumentasi dan FAQ secara dinamik  
- Future goal: gunakan LLM untuk auto-analisis trace dan tampering

# 8. Roadmap Pembangunan

- Fasa 1: tracecomp\_builder.py (CLI untuk inject)  
- Fasa 2: tracecomp\_extractor.py (CLI untuk baca)  
- Fasa 3: Plugin Red Team (GhostTag)  
- Fasa 4: Penulisan akademik, presentasi industri  
- Fasa 5: CI/CD & SBOM integration (jika traction tinggi)

# 9. Soalan Lazim & Risiko Umum

Q: Kalau metadata dibuang dari fail?  
A: SOC boleh detect absence atau gunakan fallback rule.

Q: Kalau hacker spoof ORG tag?  
A: Signature dan struktur metadata akan gagal validasi.

Q: Apa beza dengan static signature AV?  
A: Ini bukan untuk detect malware — tapi untuk kesan siapa yang buat.

# 10. Infrastructure Diagram

+------------------+  
 | Source Code |  
 +--------+---------+  
 |  
 v  
 +--------------+--------------+  
 | Compiler (e.g. GCC, Go) |  
 +--------------+--------------+  
 |  
 v  
 +------------+------------+  
 | tracecomp\_builder.py |  
 | (Inject Metadata) |  
 +------------+------------+  
 |  
 v  
 +--------+--------+  
 | Binary Output |  
 +--------+--------+  
 |  
 +--------v--------+  
 | tracecomp\_extractor.py |  
 | (Read Metadata) |  
 +------------------------+