CBMC

Verificación Formal en Software Crítico y Aplicación en Criptoanálisis

Bratti Juan, Herrador Emanuel N., Scavuzzo Ignacio

github.com/helcsnewsxd/cbmc-analysis-report/

FUNDAMENTOS Y ORIGEN

- Forma parte de la familia CPROVER
- Creado por Daniel Kroening, Edmund Clarke y Flavio Lerda
- Destinada al análisis y verificación automática de Software
- Implementa Bounded Model Checking
- Uso en la Industria + Academia

CBMC

JBMC

Kani

EBMC

INTERFAZ DE USUARIO

```
$ cbmc [options ...] file.c ...
```

```
--help (lista de comandos)
--trace (traza de ejecución)
--verbosity (nivel de detalle)
--dimacs (mostrar fórmula CNF)
--function f_name (especificar módulo)
```

```
--div-by-zero-check
--pointer-check
--bounds-check
--no-assertions-
--unwind k (unwinding a lo sumo k veces)
--unwindset L:k (para loop específico L)
```

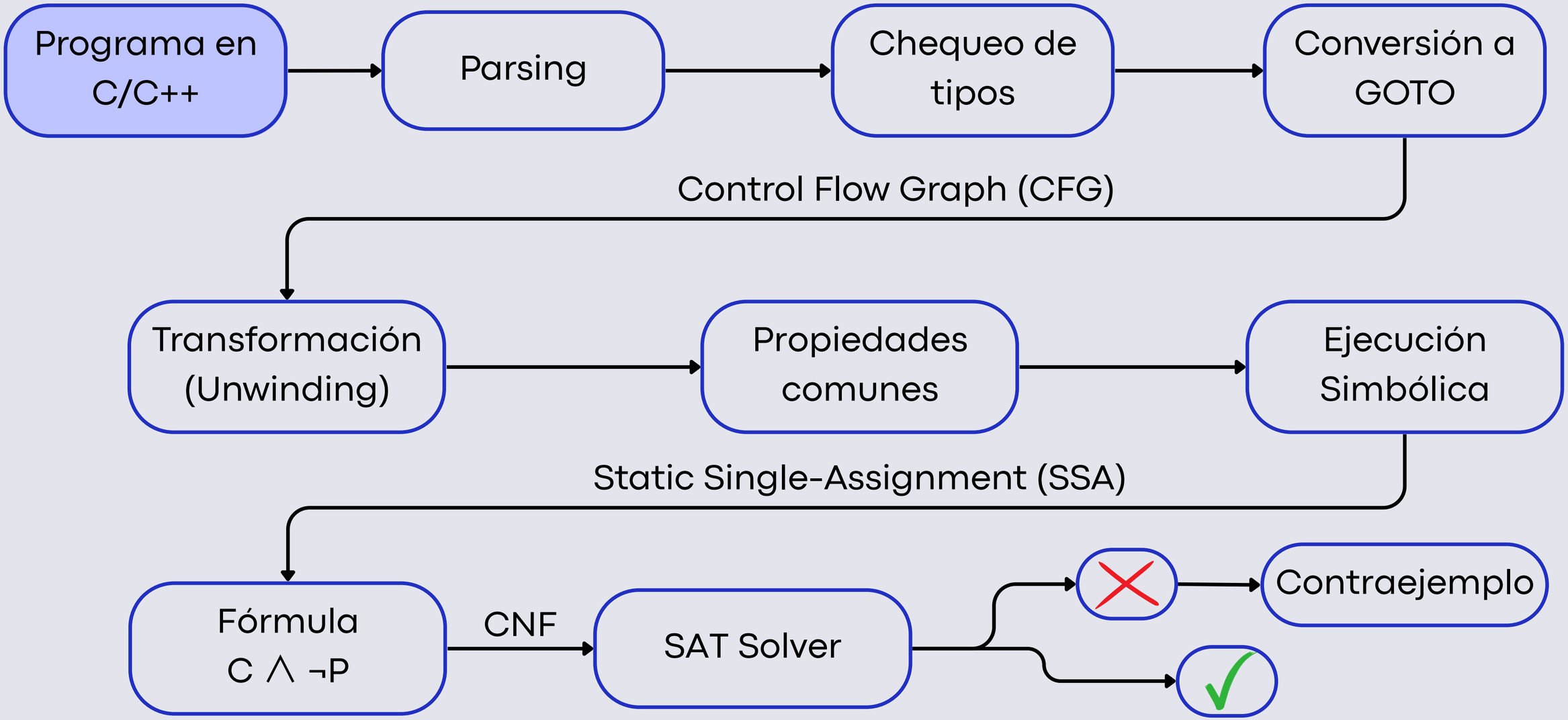
Propiedades Personalizadas

```
__CPROVER__assert(cdt, err_msg)
```

Acotar Análisis Simbólico

```
__CPROVER_assume(cdt)
```

FUNCIONAMIENTO INTERNO



ASPECTOS TÉCNICOS

```
x = x + y;
if(x != 1)
    x = 2;
else
    x++;

assert(x <= 3);
```

```
x1 = x0 + y0;
if(x1 != 1)
    x2 = 2;
else
    x3 = x1 + 1;

x4 = (x1 != 1) ? x2 : x3;
assert(x4 <= 3);
```

$$
\begin{aligned}
C :=\ & x_1 = x_0 + y_0 \ \wedge \\
& x_2 = 2 \ \wedge \\
& x_3 = x_1 + 1 \ \wedge \\
& x_4 = (x_1 \mathrel{!}= 1) \ ? \ x_2 \ : \ x_3 \\
\\
P :=\ & x_4 \leq 3
\end{aligned}
$$

APLICACIONES
Y CASOS DE ESTUDIO

VERIFICACIÓN

- Programas concurrentes
- Hardware
  - Consistencia ante fallos en memoria no volátil
- Sistemas operativos
  - TinyOS
- Implementación de drivers
- Librerías de criptografía
  - ECC
  - Pérdida de entropía en PRNGs

WCET (WORST CASE EXECUTION TIME)

BIOLOGÍA

- Restricciones del tráfico de vesículas en células eucariotas.

EQUIVALENCIA

- Entre programas
  - Generadores de código no confiables
- Simulación con modelo
  - Stateflow a C

SISTEMAS FÍSICOS Y DE CONTROL

- Interacciones de conductores con vehículos semi-autónomos

GENERACIÓN AUTOMÁTICA DE VECTORES DE PRUEBA

- Unit testing
- Coverage analysis

CASO DE USO

Reto **XtraORdinary** del CTF picoMini (2021)

01 Intercepción de mensaje hexadecimal cifrado. Encriptación XOR con clave repetida (1 clave privada, 5 públicas) al azar con las públicas.

02 Flag desconocida de longitud (hex / 2).
Key privada desconocida de longitud desconocida.
Condicionales de aplicación de encriptación a keys públicas deconocidos.

03 CBMC para búsqueda de trazas de error con aserción final que la flag encriptada es distinta a la interceptada.

04 Reducción de la búsqueda usando contratos en código, **__CPROVER_assume()**, para asegurar propiedades que la flag y key cumplen.

05 Script que itera sobre los posibles tamaños de key. **6.45**s para descifrar el mensaje.

COMPARACIÓN CON OTRAS HERRAMIENTAS

CBMC

Exploración acotada del programa

55.7%

VS

CPA-SEQ

Técnica más avanzada
Exploración profunda

59.66%

CBMC

Cotas fijas a un k

55.7%

VS

ESBMC-INER

Bounded model checking incremental

63.69%

CONCLUSIÓN

INTERFAZ	LIMITACIÓN	PROPIEDADES	APLICACIÓN
Simple y basada en la línea de comandos. Configurable mediante flags	Asegura que la propiedad se cumple o no para cierta cota	Permite verificar propiedades estándar (out of bounds, etc.) y propiedades personalizadas	Diversos usos en el ámbito académico y calidad industrial