Capturing the iccMAX Calculator Element: A Case Study on Format Design

LangSec 2022

May 26, 2022

Vijay Kothari
Prashant Anantharaman
Sean Smith
(Dartmouth College)

Briland Hitaj
Prashanth Mundkur
Natarajan Shankar
(SRI International)

Letitia W. Li
(BAE Systems)

Iavor Diatchki
William Harris
(Galois Inc.)

Talk Structure

- Background: ICC, iccMAX, DARPA SafeDocs
- Security Evaluation Method
- iccMAX Primer
- Findings & Suggestions
- General Takeaways

The Creation of the International Color Consortium

- Early 1990s:
  - Devices (e.g., scanners, printers, monitors) must communicate color information.
  - No standard exists for color management.
- 1993: eight vendors come together to create the ICC.
- Within a year, the first ICC profile specification is released.

The iccMAX Specification

- The iccMAX specification expands upon ICC v4.
- Notably, it supports the encoding of functions.
- And it...
  - was carefully designed with security/predictability in mind.
  - is relatively new.
  - is not widely adopted.
  - has an open-source reference implementation.
- We did a security evaluation of the specification.

The SafeDocs Project

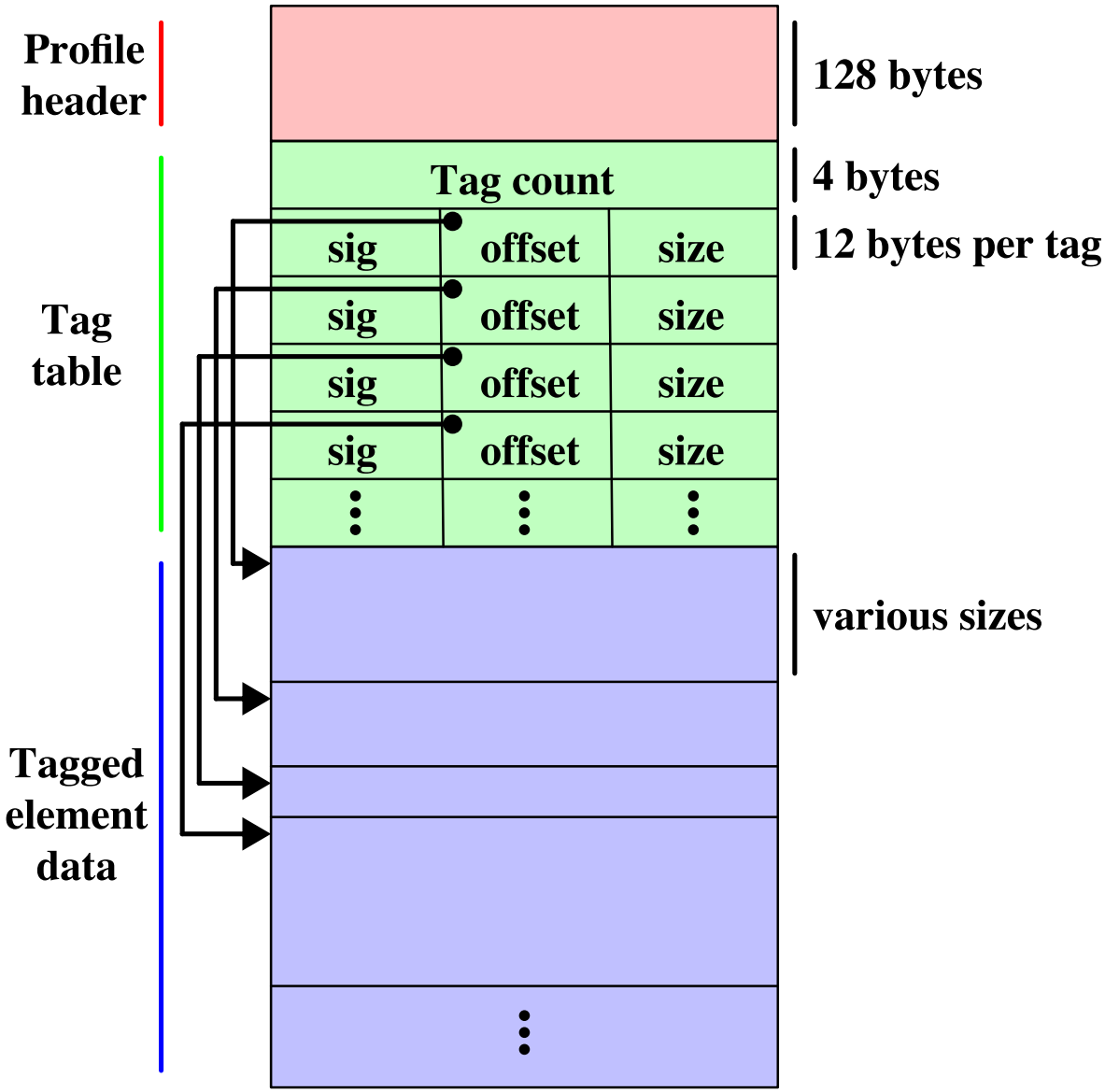
- The SafeDocs Project:
  - Problem: manually-written parsers and specification complexity breed insecurity
  - Solutions:
    - identify de facto grammars
    - identify safe subsets of them
    - create tools for safe-and-verified parsing

Security Evaluation Method

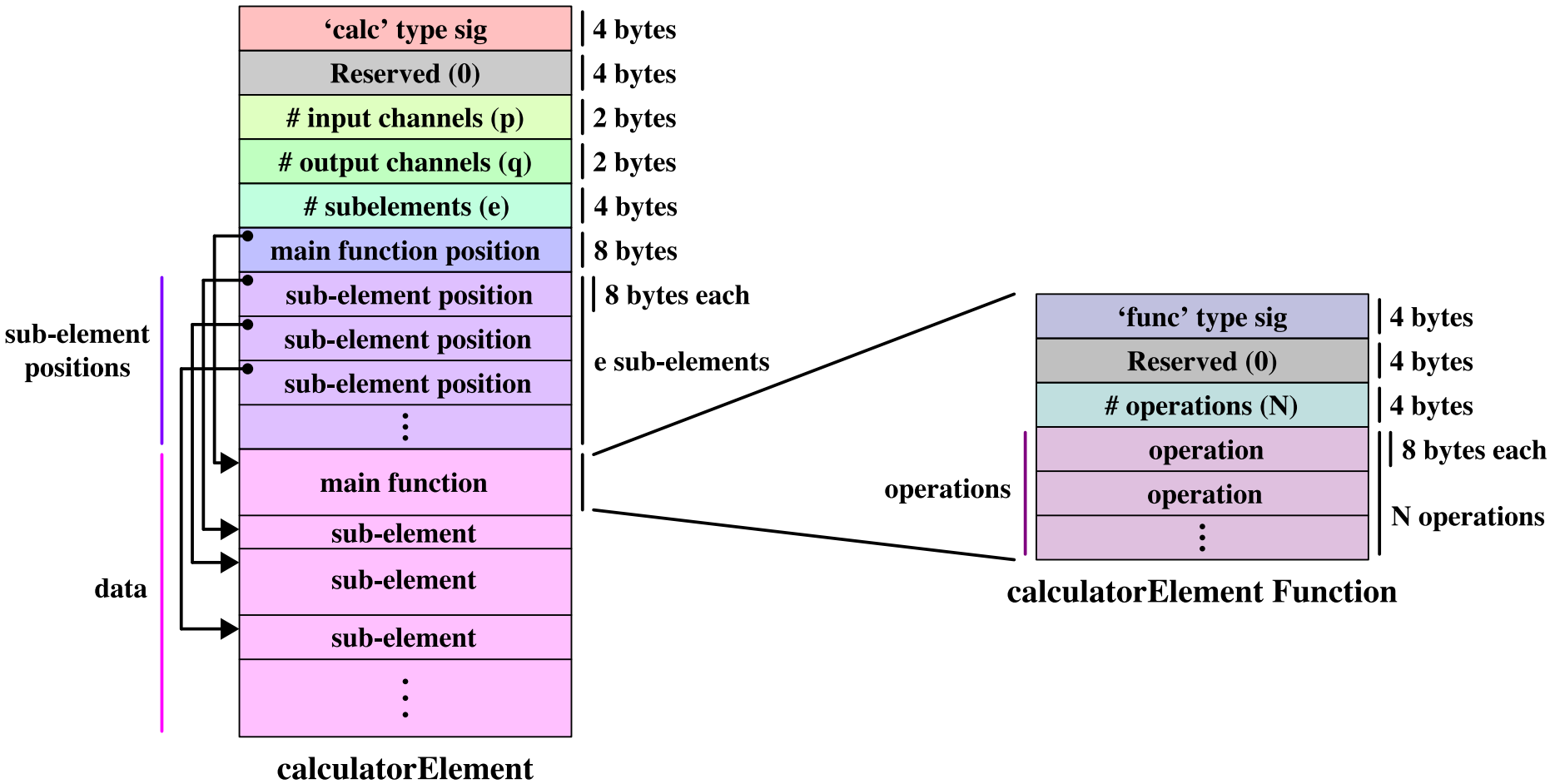
- We used various tools to conduct a security evaluation of the iccMAX specification.
  - We = {SRI/Dartmouth, BAE, Galois}
  - Tools include:
    - PVS
    - Parsley language
    - Parsley-Rust Static Analyzer
    - ACL2
    - DaeDaLus
  - Security evaluation aligns with security and predictability objectives for iccMAX. [1].

[1] https://www.color.org/whitepapers/ICC_White_Paper_52_calculatorElement_security_implementation_notes.pdf

The High-Level Structure of iccMAX Files



The calculatorElement



Findings: Completing the Resource Contract

- The calculatorElement currently specifies:
  - the number of expected input channels
  - and the number of expected output channels
- We proposed completing the resource contract w.r.t.:
  - the number of temporary channels
  - the data stack size
  - computational effort or execution cost
- Doing so would:
  - enable resource-constrained applications to quickly determine whether they can invoke a given calculator element
  - allow applications to create bounded compartments for untrusted computation

Findings: Conditional Operators are Insufficiently Defined

Should we treat nested conditional operations (and contained operations) as a single operation or multiple operations?

if 5
if 4
pi 0
pi 0
NaN 0
+INF 0

Findings: Operators Missing from the Specification

Some operations (`fJab`, `tJab`, `fLab`, and `not`) appear in the demo implementation, but they do not appear in the specification.

Findings: Non-Numeric Values Allow Parser Differentials

The specification defines exceptional values, but computation on them is left as implementation dependent.

x:=-1.0/0.0;
if x then output(1.0)
else output(2.0)

Findings: Minor Errors & Issues

- The `flip` operator (Table 96) is described to act on $S + 1$ elements but the description suggests $S + 2$ elements.

'flip' (666c6970h)	A_0 ... A_{S+1}	Reverse the top S+1 elements on the stack (T shall be zero)	A_{s+1} ... A_0

- The fourth field has field length of 8 instead of $8N$.

Table 86 — calculatorElement Function encoding

Byte position	Field length (bytes)	Content	Encoded as...
0...3	4	'func' (66756e63h) type signature	
4...7	4	Reserved, shall be 0	
8...11	4	Number of operations (N)	uInt32Number
12...12 + N*8 - 1	8	Function operations	

Updates to iccMAX

This work led to many revisions in the iccMAX specification, e.g.:

- revising select operator definitions
- adding a metadata structure after the operator table that specifies
  - position and number of sub-elements
  - maximum stack size
  - size of variables
  - maximum number of operations in code path
- adding the cost of operations to the calculatorElement metadata.
- revising the specification so that it states that floating point arithmetic should (not shall) fully implement IEEE 754

Some General Takeaways

- Format design should utilize formal methods parsing tools.
- Data should be orderly sequenced.
- Size restrictions and operations should be designed to minimize unnecessary overhead.

Thank you!

contact email address:
vijay.h.kothari@dartmouth.edu

Acknowledgements:

- DARPA SafeDocs Project.[2]
- Peter Wyatt (PDF Organization)
- The International Color Consortium
- Erik Poll and other reviewers

---

[2]Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Government or DARPA. This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR001119C0075.