

OASIS SARIF TC

Defining a standard output format for static analysis tools

Chairs: David Keaton & Luke Cartey, Chair

Secretary: Stefan Hagen, stefan@hagen.link

```
1 #include "string"
2 char secure(char * foo) {
3     char a[4];
4     strcpy(a, foo);
5     printf("%s\n", a[6]);
6     return a[0];
7 }
8 int main(int argc, char **argv) {
9     secure(argv[0]);
10 }
```

```
$ cppcheck happy_coder.cpp
```

```
Checking happy_coder.cpp ...
```

```
[happy_coder.cpp:5]: (error) Array 'a[4]' accessed at index 6, which is out of bounds.
```



Individual experts build standard in the open 1/2



Advancing open standards for the information society

[Other Languages](#) ■ [Site Map](#) ■ [Member Login](#)

I want to:  

[Standards](#) | [Committees](#) | [Join](#) | [News](#) | [Events](#) | [Resources](#) | [Member Sections](#) | [Policies](#) | [About](#)

OASIS Static Analysis Results Interchange Format (SARIF) TC

[Join This TC](#)

[TC Members Page](#)

[Send A Comment](#)

Defining a standard output format for static analysis tools

David Keaton, dmk@dmk.com, Chair

Luke Cartey, Chair

Stefan Hagen, stefan@hagen.link, Secretary

Table of Contents

- [Announcements](#)
- [Overview](#)
- [Subcommittees](#)
- [TC Liaisons](#)
- [TC Tools and Approved Publications](#)
- [Technical Work Produced by the Committee](#)



Connect with OASIS



Related links

- [Charter](#)
- [IPR Statement](#)
- [Membership](#)
- [Obligated Members](#)
- [Email Archives](#)
- [Comments Archive](#)
- [Ballots](#)
- [Documents](#)
- [Schedule](#)

Individual experts build standard in the open 2/2

- [OASIS Open Source Repositories Sponsored by the Committee](#)
- [Expository Work Produced by the Committee](#)
- [External Resources](#)
- [Mailing Lists and Comments](#)
- [Press Coverage and Commentary](#)
- [Additional Information](#)

Announcements

See press release: [Industry leaders collaborate to define SARIF interoperability standard for detecting software defects and vulnerabilities](#): Common data format for static analysis tools is being advanced by CA Technologies, Cryptsoft, FireEye, GrammaTech, Hewlett Packard Enterprise (HPE), Micro Focus, Microsoft, New Context, Phantom, RIPS, SWAMP, Synopsys, U.S. DHS, U.S. NIST, and others; 12 Oct 2017.

The first meeting of the OASIS SARIF Technical Committee was held via teleconference on September 06, 2017. David Keaton (Individual) and Luke Cartey (Semmler) were elected TC Co-Chairs.

Participation in the [OASIS SARIF TC](#) is open to all interested parties. Contact join@oasis-open.org for more information.

TC Participants

*Representing these OASIS
Foundational and Sponsors:*

CA Technologies
Cryptsoft Pty Ltd.
FireEye, Inc.
GrammaTech, Inc.
Micro Focus
Microsoft
New Context Services,
Inc.
Phantom
RIPS Technologies
Synopsys

*View full TC roster from
'Membership' link above.*

Real people and really open!

GitHub navigation bar with repository name, search, and navigation links.

Repository header for oasis-tcs / sarif-spec, including watch, star, and fork buttons, and navigation tabs for Code, Issues, Pull requests, Projects, Wiki, and Insights.

OASIS SARIF TC: Repository for development of the draft standard, where requests for modification should be made via Github Issues <https://github.com/oasis-tcs/sarif-spec>

Repository statistics: 50 commits, 4 branches, 0 releases, 3 contributors.

Repository actions: Branch: master, New pull request, Create new file, Upload files, Find file, Clone or download.

 lgolding Propose language for the result.rank property. Latest commit 2f95b28 3 days ago		
Documents	Propose language for the result.rank property.	3 days ago
meetings	Meeting minutes draft of #6 from 2017-NOV-08.	20 days ago
CONTRIBUTING.md	create static boilerplate CONTRIBUTING file	3 months ago
LICENSE.md	create static boilerplate LICENSE file	3 months ago
README.md	update with link to Issues	2 months ago
Workflow.md	Fix #67: Clarify approval process in Workflow.md	25 days ago

Purpose (from contributor slides)

Make developers more productive by enabling them to interact with results from multiple analysis tools in a uniform way.

- Enable uniform viewing experiences (*e.g.*, IDE integrations)
- Enable uniform storage in and retrieval from a back end
 - “result management systems”

Other applications (from contributor slides)

- **Dynamic analysis tools**
 - code flow support with `location.kind`
- **Web scanning tools**
 - analysis targets expressed as URLs

Design goals (from contributor slides)

- Comprehensively capture range of data produced by commonly used static analysis tools.
- Be a useful format for analysis tools to emit directly,
 - and also an effective interchange format into which the output of any analysis tool can be converted.
- Be suitable for use in a variety of scenarios related to analysis result management,
 - and be extensible for use in new scenarios.
- Reduce cost & complexity of aggregating the results of various analysis tools into common workflows.
- Capture information useful assessing project compliance with
 - corporate policy or conformance to certification standards.
- Adopt a widely used serialization format that can be parsed by readily available tools.
- Represent analysis results for all kinds of programming artifacts
 - including source code and object code.
- Represent logical construct against which a result is produced
 - such as a function, class, or namespace.
- Represent physical location at which a result is produced
 - including problems detected in nested files
 - ... such as a source file within a compressed container

History (from contributor slides)

- 2013: Originated in Microsoft's security organization
 - to unify results produced by several security-related static analysis tools.
- Developed “in the open”:
<https://github.com/sarif-standard/sarif-spec>
 - Open [issues](#) in the repo are reported in OASIS
 - for resolution by the SARIF TC –
 - including concepts from related formats such as:
 - SATE and SWAMP/SCARF.
- Supported by latest Microsoft C#/VB/C++ compilers, a variety of publicly available Microsoft tools, as well as tools from

Features (from contributor slides)

- Multiple runs per file
- Tool information
- Run/invocation information
- Rich description of “results” including:
 - “Physical” and “logical” locations
 - Multiple locations per result
 - Code flows
 - Stacks
 - Fixes
- Rule metadata
- File metadata
 - Hashes
 - MIME type
 - Embedded contents
- Support for both text and binary files
- Support for “nested” files
- Tool notifications (*e.g.* capture tool console output)
- Support for “baselining”

Example:

```
{
  "version": "1.0.0",
  "runs": [
    {
      "tool": {
        "name": "CodeScanner",
        "semanticVersion": "2.1.0"
      },
      "files": {
        "file:///user/builder/work/src/collections/list.cpp": {
          "mimeType": "text/x-c"
        }
      },
      "results": [
        {
          "ruleId": "C2001",
          "message": "Variable \"count\" was used without being initialized.",
          "locations": [
            {
              "analysisTarget": {
                "uri": "file:///user/builder/work/src/collections/list.cpp",
                "region": {
                  "startLine": 15
                }
              },
              "fullyQualifiedLogicalName": "collections::list:add"
            }
          ]
        }
      ],
      "rules": {
        "C2001": {
          "id": "C2001",
          "fullDescription": "A variable was used without being initialized. This can result in runtime errors such as null reference exceptions."
        }
      }
    }
  ]
}
```

One sample free OSS “Integrator” – SWAMP



SWAMP

SOFTWARE ASSURANCE MARKETPLACE

Do It Early. Do It Often.



Packages

Upload your code and manage your software packages.

0



Assessments

Perform assessments on packages using code analysis tools.

0



Results

View the status and results of completed assessments.

0



Runs

View assessments scheduled to run at regular intervals.

0



Projects

Create projects to share results with other users.

0



Events

View events associated with your projects & account.

2



SWAMP Funding and Support

The SWAMP is funded by the Department of Homeland Security (DHS) Science and Technology Directorate, Homeland Security Advanced Research Projects Agency, Cyber Security Division (DHS S&T/HSARPA/CSD)

Thanks.