

# Part I.

## Security analysis of MongoDB

Gardon Tomas, Jaros Miroslav, Meena Rajni

### *About project*

MongoDb is file based database engine. It's leading technology on field of NoSQL databases. It's written in C/C++ programming language, with some additional parts in JavaScript. Unlike relational databases based on SQL, mongoDB represents data as files instead of tables. For storage is used BSON data type, which is based on JSON and is often understood as Binary-jSON. Sourcecode

### *Static analysis*

Because project is very big, finding any bug in code just by reading it is almost impossible. Unfortunately despite compilable by MSVC, the mongo source is not possible to check with PreFast, because PreFast is tool dependent on compiling of source, so check with this tool was not performed. But we've succeeded with Cppcheck. The result file of cppcheck more than 1600 lines. For better understanding of bugs we will check them by severity.

<u>Severity</u>	<u>Problem type</u>	<u>First Apperance</u>	<u>count of occurences</u>
Error	ResourceLeak	db/initialize_server_global_state.cpp:173	1
	Memory leak	dbtests/preftests.cpp	2
	Uninitialized variable	util/md5.cpp	93
Warning	Member variable is not initialized in constructor	db/cloner.cpp	41
	%ld expects long but unsigned long is given	util/processinfo_linux2.cpp	1
	scanf without field limit can crash on large input data	util/net/httpclient.cpp	1

Style	class does not have constructor	bson/bson_validate.cpp:148	12
	the scope of variable can be reduced	client/connpool.cpp:124	33
	Exception should be caught by reference	client/connpool.cpp:129	5
	Variable is assigned value but never used	db/repl/master_slave.cpp:1316	15
Performance	Prefer Prefix ++/-- operators for non-primitive types	client/connpool.cpp	Many times, everywhere iterator is used

Cppcheck also shown some things with macro definitions, but those are not relevant, because cppcheck is only telling that it doesn't know what it will do :)

### *Understanding reported issues*

Since MongoDB is one of the most used NoSQL databases, it's no surprise that there are not many severe bugs or vulnerabilities. The fact that it's open-source project helps that.

Resource leak: False positive, Cppcheck badly understands usage of freopen function, which

So it means, that during assignment the file handler will be rewritten and the opened file will remain lost.

Memory leak: False positive

Uninitialized Variable: False positive, this error is present on almost hundred lines of code in row,

reason is complicated code in this part. The variable is covered by many macros, and many macros appears on lines below, so that code is very hardly readable and understandable.

Member variable is not initialized in constructor: False positive

%ld expects long but unsigned long is given: Might be problem

scanf without field limit can crash on large input data: **Possible vulnerability**

The function is used probably to check validity of obtained HTTP header.

If the input received would be word longer than 32bytes followed by integer, the buffer overflow would appear and might cause stack buffer overflow.

Style issues:

The style issues are not in any means problematic with exception related to exceptions

Generally in C++ every exception should be caught by reference, catching by value is generally considered as Invalid and potentially dangerous.

### *Result*

MongoDB is very known and very controlled piece of software, because many of companies and services relies on solidity and stability of this database engine. That's why it's expected to be as buggy free as possible.