KWUN BHANSALI LAZARUS LLP
MICHAEL S. KWUN (SBN 198945)
mkwun@kblfirm.com
555 Montgomery St., Suite 750
San Francisco, CA 94111
Telephone: 415 630-2350
Facsimile: 415 367-1539

WUERSCH & GERING LLP
V. DAVID RIVKIN (admitted *pro hac vice*)
david.rivkin@wg-law.com
JOHN A. SMITTEN (admitted *pro hac vice*)
john.smitten@wg-law.com
100 Wall St., 10th Fl.
New York, NY 10005
Telephone: 212 509-5050
Facsimile: 212 509-9559

Attorneys for Defendant
FLORAGUNN GmbH

UNITED STATES DISTRICT COURT

NORTHERN DISTRICT OF CALIFORNIA

OAKLAND DIVISION

ELASTICSEARCH, INC., a Delaware corporation, ELASTICSEARCH B.V., a Dutch corporation, Plaintiffs, v. FLORAGUNN GmbH, a German corporation, Defendant.	Case No. 4:19-cv-05553-YGR **ANSWER TO FIRST AMENDED COMPLAINT WITH COUNTERCLAIMS**

Defendant floragunn GmbH ("floragunn") answers the Amended Complaint of Elasticsearch, Inc. and Elasticsearch B.V. (together "Elastic") filed on November 26, 2019, as follows:

PRELIMINARY STATEMENT

This case is about a large public company, Elastic (NYSE: ESTC), seeking to critically damage a small competitor, floragunn, by making false accusations of copyright infringement related to floragunn's only product -- a security "plugin" that floragunn developed for use with Elastic's popular Elasticsearch search engine and Kibana software. Elasticsearch is a search and analytics engine that allows users to build upon it and search out their own data. Elasticsearch may be accessed through various programing languages, including Java, Python, JavaScript, a REST API and others. Kibana is Elastic's user interface designed to manage and configure Elasticsearch and other Elastic products and to produce data visualizations including diagrams and dashboards. Elastic does not allege that floragunn's plugin infringes either Elasticsearch or Kibana. Indeed, Elastic encourages programmers to develop plugins to enhance the core functionality of Elasticsearch and Kibana. Rather, Elastic alleges that floragunn's security plugin infringes Elastic's own security plugin for Elasticsearch and Kibana.

Elastic's allegations of infringement are meritless. floragunn's security plugin for Elasticsearch (called "Search Guard") was developed before Elastic created its own security plugin product called Elastic Shield ("Shield"). When Elastic released Shield in 2015, Elastic and floragunn became competitors in the security plugin market. Rather than compete fairly with floragunn's product, Elastic commenced this action as part of its ongoing scheme to cause fear, uncertainty, and doubt among floragunn's customers and potential customers, irrespective of the lack of truth to its claims of infringement.

Elastic has also contacted floragunn's existing customers directly to tell them, falsely, that they face legal exposure if they continued using floragunn's product, and offering Elastic's own security plugin as a "safe" substitute. Elastic has also posted false statements on Elastic's blog and elsewhere. In addition, Elastic's sales representatives have made false statements to

floragunn's clients and prospective clients about the origins of floragunn's funding, and falsely claiming that former Elastic employees stole Elastic's code and gave it to floragunn.

Unfortunately for floragunn, Elastic's scheme has already begun to work. Several of floragunn's customers have chosen not to renew their existing licenses from floragunn at a rate that is substantially higher than at any time in the floragunn's history, floragunn's revenues have begun to decline, and an unusually high number of floragunn's customers are debating whether to renew their licenses.

Yet as further set out in floragunn's defenses below, floragunn's source code was independently created. floragunn did not copy Elastic's source code for its security plugin. Moreover, Elasticsearch and Kibana are themselves based on code not original to Elastic, including but not limited to Lucene, Netty, AngularJS, Lodash, and Node.js. Many aspects of an Elasticsearch or Kibana plugin are constrained by choices made by the programmers who wrote that code, by the need to function as a plugin to Elasticsearch or Kibana, or by other factors that limit the scope of copyright protection. Like Elasticsearch and Kibana, aspects of Shield are based on code not original to Elastic, including but not limited to open source libraries or code such as Lucene, Netty, AngularJS, Lodash, and Node.js., and therefore are not Elastic's original expression or are otherwise not entitled to copyright protection.

Other aspects of the code at issue concern standard, common, or stock programming practices. floragunn denies copying of any protectable expression original to Elastic.

ANSWERS TO SPECIFIC ALLEGATIONS IN THE AMENDED COMPLAINT

floragunn answers each of Elastic's specific allegations in the Amended Complaint as follows:

Paragraph 1

Allegation: Elasticsearch, Inc. and elasticsearch B.V. (collectively "Elastic") bring this action to remedy floragunn GmbH's ("floragunn") knowing and

willful infringement of Elastic's copyright in the source code for Elastic's X-Pack software.

Response: floragunn denies Elastic's allegations set forth in paragraph 1 of the Amended Complaint, and further states that it has not infringed any of plaintiff's source code.

Paragraph 2

Allegation: Elastic is the creator of the Elastic Stack suite of products that is centered on the popular and powerful Elasticsearch search and analytics engine. Leading companies and organizations like Cisco Systems, Facebook, and NASA's Jet Propulsion Laboratory at the California Institute of Technology use and depend upon Elasticsearch.

Response: floragunn denies having specific knowledge of whether Elastic is the "creator" of the "Elastic Stack" or whether Elasticsearch and whether the companies cited "use and depend" on Elasticsearch, and therefore denies knowledge or information sufficient to respond to Elastic's allegations set forth in paragraph 2 of the amended complaint, and therefore denies such allegations.

Paragraph 3

Allegation: Elastic offers a set of features, known as X-Pack, that enhance and extend the Elastic Stack suite of products. In keeping with its longstanding commitment to openness, Elastic made the source code for X-Pack publicly available in 2018 subject to certain restrictions. Among other rights, Elastic clearly reserved commercial rights in X-Pack and its derivative works.

Response: (1) floragunn admits the allegations in the first sentence of paragraph 3 of the amended complaint. (2) As to the second sentence of paragraph 3, as described in more detail in response to the allegations, floragunn denies that Elastic has a "longstanding commitment to openness," since the source code of the commercial parts of the Elastic Stack (formerly known as Shield, Watcher, and X-Pack among other names) were closed to the public from their initial release in January 2015 until April 2018. (3) As to the third sentence of paragraph 3, floragunn denies that "Elastic clearly reserved commercial rights in X-pack and its derivative works," and refers the Court to its response to the allegations to paragraph 20 of the amended complaint below.

Paragraph 4

> Allegation: floragunn markets and distributes Search Guard, a plug-in for Elasticsearch that is intended to compete with the security features of X-Pack. Yet instead of fairly competing with Elastic and developing Search Guard with its own resources, floragunn copied multiple and critical portions of Elastic's X-Pack proprietary security source code into its Search Guard product.

Response: (1) floragunn denies Elastic's allegations in the first sentence of paragraph 4 of the amended complaint, and specifically notes that floragunn's security plugin products (Search Guard and its predecessors ESP and Defender), were created and made available to the public as open code as early as 2013, years before Elastic's security solution was ever made available. (2) floragunn denies Elastic's allegations in the second sentence of paragraph 4 of the amended complaint, and specifically avers that at no time did floragunn "cop[y] multiple critical portions of Elastic's X-Pack proprietary security" source code into Search Guard or any other product.

Paragraph 5

> Allegation: One particularly large incident of copying occurred just one month after Elastic publicly opened X-Pack's source code. Elastic's examination of floragunn's then-publicly available code on GitHub demonstrates that, at that time, floragunn made dramatic alterations to Search Guard in a single, massive effort that it released—contrary to common programming practice and floragunn's own past practices—without any substantive explanation.

Response: floragunn denies that it "copied" or "infringed" Elastic's code and therefore denies Elastic's allegations in paragraph 5 of the amended complaint. floragunn further states that the alterations to floragunn's code were neither dramatic, nor contrary to common practices, and were driven by a client's request, which is fully documented.

Paragraph 6

> Allegation: But this was not the beginning or the end of floragunn's infringement. Elastic has now discovered evidence that floragunn's copying and creation of derivative works from Elastic's code extends back to at least 2015. Because Elastic had released that code only in binary form, moreover, it was necessary for floragunn to intentionally decompile that code to enable the copying and creation of derivative works. Furthermore, Elastic has now determined that

> floragunn copied and created derivative works not only from Elastic's X-Pack code containing security features for its Elasticsearch software—floragunn also copied and created derivative works from Elastic's X-Pack code containing security features for its Kibana software.

Response: floragunn denies Elastic's allegations in paragraph 6 of the amended complaint. Furthermore, Elastic's claim that it "has now discovered evidence" (emphasis added) that floragunn's alleged copying of Elastic's code "extends back to at least 2015" is particularly disingenuous since floragunn's source code has been available and open for public inspection, including inspection by Elastic, continuously since 2015, and Shay Banon, the founder of Elastic, has previously (as early as 2016) made unsubstantiated assertions to floragunn that Search Guard somehow violated Elastic's IP, but when pressed for specifics, Elastic never provided any. The fact that Elastic has been analyzing the source code for Search Guard for over four years, but has only been able to identify approximately 100 lines out of more than 60,000 lines of code that allegedly have some similarity speaks volumes about the Elastic's true intention in commencing this lawsuit.

Paragraph 7

> Allegation: Once floragunn copied Elastic's code, it then licensed its infringing Search Guard software to corporations and institutions, including a significant number that are located in the Northern District of California. These acts by floragunn induced further infringements of Elastic's copyrights by those third parties, including through products and services offered by those third parties. For example, Elastic has now determined that the Amazon Elasticsearch Service and Open Distro for Elasticsearch from Amazon.com, Inc. and Amazon Web Services, Inc., as well as Rackspace US, Inc.'s ObjectRocket for Elasticsearch and IBM Corporation's Cloud Databases for Elasticsearch include and/or recently included infringing code.

Response: floragunn denies that it "copied" or "infringed" Elastic's code, and that a significant number of its clients are located in this District, and therefore denies Elastic's allegations in paragraph 7 of the amended complaint.

Paragraph 8

> Allegation: floragunn's response to Elastic's infringement claims is also consistent with copyright infringement by floragunn. After the commencement of this lawsuit, Elastic issued takedown notices under the Digital Millennium Copyright Act ("DMCA") to websites that were hosting floragunn's infringing code. Those websites removed floragunn's code in response to Elastic's notices. floragunn had the right to issue counter notifications to those websites to assert that floragunn was, contrary to Elastic's notices, the owner of the copyright to the code in question. But Elastic has seen no such notices from floragunn because, on information and belief, floragunn issued no such notices. But what floragunn did do is telling: it moved hosting for downloads of its infringing code to a hosting provider that expressly advertises that it will not comply with the DMCA. The hosting provider's website states: "Purchasing USA-based hosting for a site that is not legal to be run in America is not a sensible thing to do. Offshore hosting can be helpful for less scrupulous businesses who wish to bypass local laws or regulations, particularly for issues like copyright law, which is also known as no DMCA hosting."

Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 8 of the amended complaint. floragunn further states that Elastic mischaracterizes what the DMCA process is, and why floragunn moved its source code to an off-shore host after GitHub was served with a DMCA notice, and refers the Court to floragunn's response to paragraph 76 of the amended complaint, below.

Paragraph 9

> Allegation: floragunn's unauthorized reproduction, creation of derivative works, and distribution of Elastic's copyrighted software code constitutes copyright infringement under 17 U.S.C. § 101 *et seq*. floragunn is further liable for contributory copyright infringement because it intentionally induced Search Guard users and third parties that integrate Search Guard code into their own products and services to infringe Elastic's copyrights. Elastic seeks injunctive and monetary relief to the maximum extent permitted by law.

Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 9 of the amended complaint.

Paragraph 10

> Allegation: Plaintiff Elasticsearch, Inc. is incorporated in Delaware; it has its principal place of business in Mountain View, California. Plaintiff elasticsearch B.V. is incorporated in the Netherlands.

Response: floragunn denies having knowledge or information sufficient to respond to Elastic's allegations set forth in paragraph 10 of the amended complaint, and therefore denies such allegations.

Paragraph 11

> Allegation: Defendant floragunn is a German company with a principal place of business in Berlin, Germany.

Response: floragunn admits the allegations in paragraph 11 of the amended complaint.

Paragraph 12

> Allegation: Elastic is aware that there are third party users and adopters of floragunn's infringing Search Guard product and code. Elastic may seek leave to amend to add those third parties as defendants following discovery from floragunn.

Response: floragunn denies that Search Guard is an "infringing product" and therefore denies Elastic's allegations in paragraph 12 of the amended complaint.

Paragraph 13

> Allegation: Elastic's claims for copyright infringement arise under the Copyright Act of 1976, 17 U.S.C. § 101 *et seq.*

Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 13 of the amended complaint.

Paragraph 14

> Allegation: This Court has original subject matter jurisdiction of this action under 28 U.S.C. §§ 1331 and 1338.

Response: floragunn admits that this Court has subject matter jurisdiction concerning the claims made by Elastic in this case, but denies that it has engaged in any action in violation of the Copyright Act of 1976.

Paragraph 15

> Allegation: This Court has specific personal jurisdiction over floragunn because, among other reasons, floragunn has extensively offered and distributed

> its infringing product containing Elastic's copyrighted material to companies in California and purposefully committed within California the acts upon which Elastic's claims arise. Additionally, to the extent floragunn has committed the illegal acts described herein outside of California, it did so knowing and intending that such acts would cause injury to Elastic at its principal place of business within California.

Response: floragunn denies Elastic's allegations in paragraph 15 of the amended complaint, but does not challenge the Court's personal jurisdiction over floragunn in connection with this litigation.

Paragraph 16

> Allegation: Venue is proper in the Northern District of California under 28 U.S.C. § 1391(b)(2) and 1391(c)(3) because a substantial part of the events or omissions giving rise to the claims alleged in this complaint occurred in this judicial district.

Response: floragunn denies that it has engaged in any acts or omissions "giving rise to the claims alleged in the complaint" but does not challenge that venue is proper in this judicial district.

Paragraph 17

> Allegation: Because this action arises from Elastic's assertion of its intellectual property rights, Northern District of California Local Rule 3.2(c) excludes this action from the division-specific venue rule and subjects this action to assignment on a district-wide basis.

Response: floragunn admits the allegation in paragraph 17.

Paragraph 18

> Allegation: Elastic produces a core suite of search and analytics products known as Elastic Stack (formerly known as ELK Stack). The Elastic Stack consists of Elasticsearch, Logstash, Kibana, and Beats. Elasticsearch is a search and analytics engine. Logstash is a server-side data processing pipeline that ingests data from multiple sources simultaneously, transforms it, and then sends it to a "stash" like Elasticsearch. Kibana lets users visualize data with charts and graphs in Elasticsearch. Beats is a family of "data shipper" software that collects and centralizes data that feeds into the other products in Elastic Stack.

Response: floragunn admits the allegations in paragraph 18 of the amended complaint.

Paragraph 19

> Allegation: X-Pack is a set of add-on features to Elastic's core Elastic Stack suite of products. X-Pack includes security, alerting, monitoring, reporting, and other add-ons to Elasticsearch, Kibana, and other products in the Elastic Stack. The predecessor to much of X-Pack was known as Shield. Elastic refers to Shield and X-Pack collectively herein as "X-Pack."

Response: (1) floragunn admits the allegations in the first sentence of paragraph 18, (2) but denies Elastic's allegations in the second sentence in paragraph 19 that the "predecessor of much of X-Pack was known as Shield." Shield was only one part of X-Pack (the security part). According to Elastic's own description, in addition to security X-Pack included "alerting, monitoring, reporting, graphic analytics, dedicated APM UI's and machine learning."

Paragraph 20

> Allegation: Elastic has a longstanding commitment to opening the source code underlying many of its products in order to facilitate building a community that helps improve and advance Elastic's products to produce the best software possible. When Elastic releases the source code for its software, it does so under clearly delineated conditions.

Response: (1) floragunn denies the allegations in paragraph 20 of the amended complaint. Specifically, Elastic misleads when it claims that "it has a longstanding commitment to opening source code underlying many of its products," since it has no "longstanding" commitment to opening source in the case of X-Pack (including Shield) because the code for X-Pack was closed source software from the time it was first released in 2015 until Elastic finally opened the source code to the public in 2018. Before April 2018, it was impossible for third-party developers to contribute anything to the proprietary and closed-source X-Pack code. Search Guard, on the other hand, has been publicly available open code since it was first released in 2015, and its predecessor ESP since 2013. (2) Second, it is false that "When Elastic releases the source code for its software, it does so under clearly delineated conditions." For example, Elastic released

both its Apache 2 licensed code and its own Elastic licensed code in the same GitHub repository, causing significant confusion as to which license applied to which files. This practice is commonly referred to as "code mingling" and is highly discouraged by the Open Source community because it leads to situations where a single commit by a developer could contain both Apache2 and Elastics licensed code. Such commits are called "toxic" for obvious reasons.

Paragraph 21

> Allegation: In late April 2018, Elastic opened the source code for version 6.2.x of X-Pack. Elastic made the code available on Elastic's public GitHub code repository for users to inspect, contribute, create issues, and open pull requests, all pursuant to the "Elastic License." Elastic has released the source code for subsequent versions of X-Pack on GitHub, also under the "Elastic License."

Response: floragunn admits Elastic's allegation in the first sentence of paragraph 21 of the amended complaint that Elastic opened the source for version 6.2x of X-Pack in April 2018, but denies that Elastic made the code available solely pursuant to the "Elastic License." floragunn denies all other allegations in paragraph 21 of the amended complaint.

Paragraph 22

> Allegation: The Elastic License did not grant to floragunn or any other party the right to create copies or prepare derivative works for use in any production capacity. And to the extent floragunn acquired any rights pursuant to the Elastic License, those rights terminated immediately and automatically by virtue of floragunn's breaches as described herein. Nor did any license applicable to earlier versions of X-Pack and/or Shield provide floragunn the right to create copies or prepare derivative works for use in any production capacity.

Response: floragunn denies the allegations of paragraph 22 of the amended complaint because the allegation implies that floragunn copied or prepared derivative works of X-Pack or Shield, which it did not. As for the legal interpretation of Elastic's licenses, floragunn respectfully refers the Court to Elastic's license to ascertain its terms.

Paragraph 23

> Allegation: floragunn markets and distributes Search Guard, a plug-in for Elasticsearch that offers features similar to the security features that Elastic offers in X-Pack. floragunn makes the source code for Search Guard available

> for review and inspection on its GitLab repositories under several different license agreements. (Before the commencement of this lawsuit, floragunn made the source code for Search Guard available through GitHub repositories.)

Response: floragunn denies Elastic's allegations in the first sentence of paragraph 23 that floragunn's "*Search Guard, a plug-in for Elasticsearch . . . offers features similar to the security features that Elastic offers in X-Pack.*" In fact, Search Guard always has, and continues to, offer more and different features than the X-Pack security plugin, although some features are similar.

Paragraph 24

> Allegation: Search Guard is available as a "Community Edition" for free for certain uses, but floragunn charges customers for Enterprise and Compliance editions of Search Guard. floragunn prohibits users from, among other things, taking features from the Enterprise or Compliance editions of Search Guard into production without purchasing a license. In fact, floragunn explicitly warns its users that doing so "is illegal" and "can lead to serious legal consequences, which can bring more harm and costs to a company . . . ."

Response: floragunn admits the allegations in paragraph 24 of the amended complaint. Search Guard community edition is available free of charge, and the Enterprise and Compliance editions are available for a fee under different licenses. floragunn further states that it did not violate the terms of any Elastic license.

Paragraph 25

> Allegation: Elastic is informed and believes, and, on that basis, alleges that after Elastic made the source code for X-Pack version 6.2.x publicly available, floragunn accessed significant portions of at least the version 6.2.x code, copied and/or created derivative works from that code, and reproduced and distributed it in the code for Search Guard.

Response: floragunn denies Elastic's allegations in paragraph 25 of the amended complaint.

Paragraph 26

> Allegation: On June 7, 2018, just over one month after Elastic made the source code for X-Pack version 6.2.x publicly available under the Elastic License, floragunn made a sudden and very large change to the Search Guard code. This change comprised 244 additions and 145 deletions of code. Many of these

> changes involved the wholesale copying of the X-Pack code that Elastic opened little over a month before.

Response: floragunn denies that it had engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 26 of the amended complaint. floragunn denies that the commit referred to in paragraph 26 of the amended complaint was sudden, very large, or precipitated by the release of X-Pack version 6.2.x. floragunn denies that the changes the referenced commit involved wholesale copying of X-Pack code.

Paragraph 27

> Allegation: A significant portion of floragunn's copying centered on the Document Level Security ("DLS") features in Elastic's X-Pack code. As the name would suggest, DLS allows an X-Pack customer to apply security settings to particular documents in the database.

Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 27 of the amended complaint. floragunn further states that, contrary to Elastic's assertions in the original complaint, which Elastic has removed from its amended complaint, the code "for computing the number of documents for DLS" is not "unique."

Paragraph 28

> Allegation: As part of its June 7, 2018, changes, floragunn copied the implementations of at least two methods from the X-Pack code, getLiveDocs and numDocs, from the file DocumentSubsetReader.java.

Response: floragunn denies that it has engaged in any copyright infringement or that it copied any implementation of Elastic's code, and therefore denies Elastic's allegations in paragraph 28 of the amended complaint.

Paragraph 29

> Allegation: A comparison of Elastic's implementation of getLiveDocs in X-Pack and floragunn's implementation of method getLiveDocs in Search Guard shows that floragunn's implementation is substantively identical to Elastic's implementation:

Elastic's Implementation of getLiveDocs:

```
@Override
public Bits getLiveDocs() {
final Bits actualLiveDocs = in.getLiveDocs();
if (roleQueryBits == null) {
// If we would a <code>null</code> liveDocs then that would mean that no
        docs are marked as deleted,
   // but that isn't the case. No docs match with the role query and therefor all
        docs are marked as deleted
     return new Bits.MatchNoBits(in.maxDoc());
    } else if (actualLiveDocs == null) {
   return roleQueryBits;
 } else {
   // apply deletes when needed:
   return new Bits() {
     @Override
     public boolean get(int index) {
        return roleQueryBits.get(index) && actualLiveDocs.get(index);
     }
     @Override
     public int length() {
        return roleQueryBits.length();
     }
   };
 }
}
```

floragunn's Implementation of getLiveDocs:

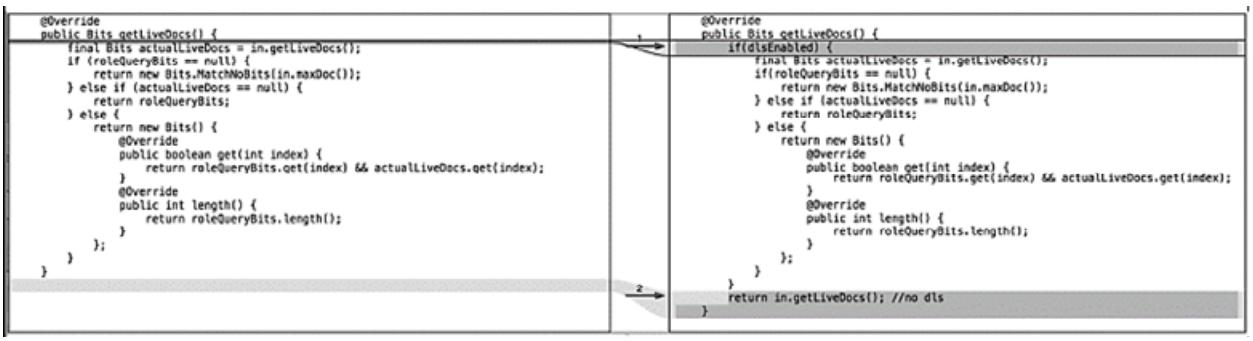
```
@Override
public Bits getLiveDocs() {
if(dlsEnabled) {
   final Bits currentLiveDocs = in.getLiveDocs();
   if(bs == null) {
                return new Bits.MatchNoBits(in.maxDoc());
   } else if (currentLiveDocs == null) {
     return bs;
     } else {
       return new Bits() {
          @Override
          public boolean get(int index) {
             return bs.get(index) && currentLiveDocs.get(index);
          }
          @Override
          public int length() {
          return bs.length();
          }
       };
    }
```

```
            }
            return in.getLiveDocs(); //no dls
        }
```

Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 29 of the amended complaint. floragunn further denies that its implementation is substantively identical to Elastic's implementation. In addition, floragunn further specifically denies that any similarities are due to copying of any protectable expression original to Elastic.

Paragraph 30

> Allegation: By removing comments and superfluous blank lines, and by making variable names consistent, it becomes apparent that the Search Guard code is copied from or is, at least, a derivative work of Elastic's code. (Elastic's code is on the left; floragunn's is on the right.) A larger version of this graphic is attached to this Complaint as Exhibit A.



```
@Override
public Bits getLiveDocs() {
    final Bits actualLiveDocs = in.getLiveDocs();
    if (roleQueryBits == null) {
        return new Bits.MatchNoBits(in.maxDoc());
    } else if (actualLiveDocs == null) {
        return roleQueryBits;
    } else {
        return new Bits() {
            @Override
            public boolean get(int index) {
                return roleQueryBits.get(index) && actualLiveDocs.get(index);
            }
            @Override
            public int length() {
                return roleQueryBits.length();
            }
        };
    }
}
```

```
@Override
public Bits getLiveDocs() {
    if(dlsEnabled) {
        final Bits actualLiveDocs = in.getLiveDocs();
        if(roleQueryBits == null) {
            return new Bits.MatchNoBits(in.maxDoc());
        } else if (actualLiveDocs == null) {
            return roleQueryBits;
        } else {
            return new Bits() {
                @Override
                public boolean get(int index) {
                    return roleQueryBits.get(index) && actualLiveDocs.get(index);
                }
                @Override
                public int length() {
                    return roleQueryBits.length();
                }
            };
        }
    }
    return in.getLiveDocs(); //no dls
}
```

Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 30 of the amended complaint. floragunn further denies that the changes Elastic proposes in paragraph 30 render anything "apparent." floragunn specifically denies that the side-by-side comparison in paragraph 30, which Elastic created by deleting code text that appears in floragunn's code and inserting code text that does not appear in floragunn's code, as well as re-formatting floragunn's source code text, supports the conclusions Elastic purports to make. floragunn notes that Elastic's changes to floragunn's actual code include:

a. Changed this: final Bits currentLiveDocs = in.getLiveDocs()

To this: final Bits actualLiveDocs = in.getLiveDocs()

b. Changed this: if(bs == null) }

To this: if(roleQueryBits == null) {

c. Changed this: } else if (currentLiveDocs == null {
return bs;

To this: } else if (actualLiveDocs == null) {
return roleQueryBits;

d. Changed this: return bs.get(index) && currentLiveDocs.get(index);

To this: return roleQueryBits.get(index) && actualLiveDocs.get(index);

e. Changed this: return bs.length();

To this: return roleQueryBits.length();

In short, the code on the "floragunn" side of the side-by-side comparison in paragraph 30 doesn't actually exist.

Paragraph 31

Allegation: Similarly, floragunn's June 7, 2018 commit changed Search Guard's implementation of the method numDocs to be essentially identical to Elastic's implementation in X-Pack. Here is Elastic's implementation, again from the file DocumentSubsetReader.java:

```
@Override
public int numDocs() {
    // The reason the implement this method is that numDocs should be
  equal to the number of set bits in liveDocs. (would be weird otherwise)
     // for the Shield DSL use case this get invoked in the QueryPhase
class (in core ES) if match_all query is used as main query
   // and this is also invoked in tests.
   if (numDocs == -1) {
     final Bits liveDocs = in.getLiveDocs();
     if (roleQueryBits == null) {
       numDocs = 0;
     } else if (liveDocs == null) {
       numDocs = roleQueryBits.cardinality();
     } else {
       // this is slow, but necessary in order to be correct:
       try {
```

```
            DocIdSetIterator iterator = new FilteredDocIdSetIterator(new
        BitSetIterator(roleQueryBits, roleQueryBits.approximateCardinality())) {
                    @Override
                            protected boolean match(int doc) {
                       return liveDocs.get(doc);
                    }
                };
                int counter = 0;
                        for (int docId = iterator.nextDoc(); docId <
          DocIdSetIterator.NO_MORE_DOCS; docId = iterator.nextDoc()) {
                    counter++;
                }
                numDocs = counter;
            } catch (IOException e) {
                        throw ExceptionsHelper.convertToElastic(e);
            }
            }
          }
          return numDocs;
        }
```

Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 31 of the amended complaint. floragunn further denies that its implementation of numDocs is "essentially identical" to Elastic's implementation. floragunn specifically denies that any similarities are due to copying of any protectable expression original to Elastic.

Paragraph 32

Allegation: Again, floragunn's June 7, 2018, changes altered Search Guard's implementation of the method numDocs to be substantively identical to Elastic's implementation in X-Pack:

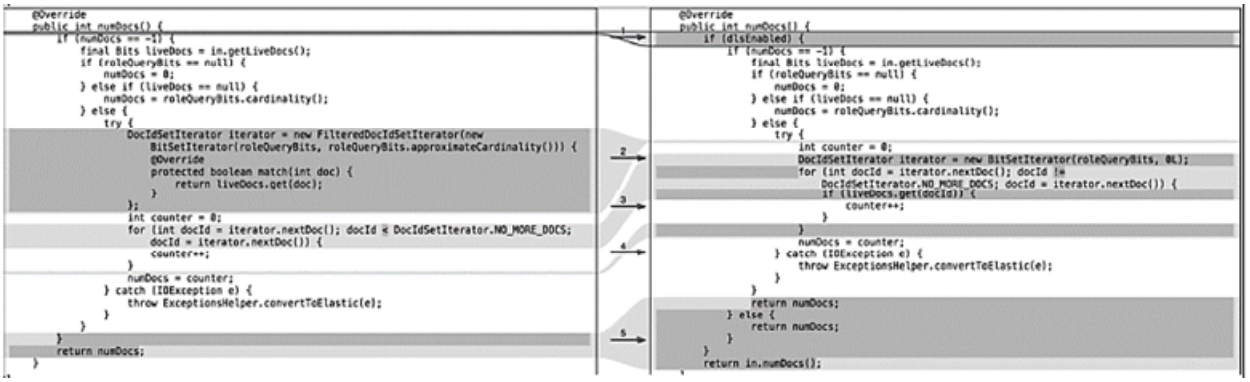
```
@Override
public int numDocs() {
    if (dlsEnabled) {
      if (this.numDocs == -1) {
        final Bits currentLiveDocs = in.getLiveDocs();
        if (bs == null) {
            this.numDocs = 0;
        } else if (currentLiveDocs == null) {
            this.numDocs = bs.cardinality();
        } else {
            try {
```

```
                int localNumDocs = 0;
                DocIdSetIterator it = new BitSetIterator(bs, 0L);
                        for (int doc = it.nextDoc(); doc !=
            DocIdSetIterator.NO_MORE_DOCS; doc = it.nextDoc()) {
                            if (currentLiveDocs.get(doc)) {
                        localNumDocs++;
                    }
                    }
            this.numDocs = localNumDocs;
            } catch (IOException e) {
                    throw ExceptionsHelper.convertToElastic(e);
                    }
                }
                return this.numDocs;
              } else {
                return this.numDocs; // cached
             }
           }
          return in.numDocs();
```

Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 32 of the amended complaint. floragunn further denies that its implementation of numDocs is "substantively identical" to Elastic's implementation. floragunn specifically denies that any similarities are due to copying of any protectable expression original to Elastic.

Paragraph 33

Allegation: Ignoring non-substantive differences in the code (*i.e.*, removing blank lines, conforming variable names, and removing the superfluous "this." in front of certain variables), it is clear that the floragunn code (on the right) is copied from or, at least, a derivative work of the Elastic code (on the left). A larger version of this graphic is attached to this Complaint as Exhibit B.



Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 33 of the amended complaint. floragunn further denies that the changes Elastic proposes in paragraph 33 render anything "apparent." floragunn specifically denies that the side-by-side comparison in paragraph 33, which Elastic created by deleting code text that appears in floragunn's code and inserting code text that does not appear in floragunn's code, as well as re-formatting floragunn's source code text, supports the conclusions Elastic purports to make. floragunn notes that Elastic's changes to floragunn's actual code include:

a. Changed this: `if (this.numDocs == -1) {`
to this: `if (numDocs == -1) {`

b. Changed this: `final Bits currentLiveDocs = in.getLiveDocs();`
to this: `final Bits liveDocs = in.getLiveDocs();`

c. Changed this: `if (bs == null) {`
to this: `if (roleQueryBits == null) {`

d. Changed this: `this.numDocs = 0;`
to this: `numDocs = 0;`

e. Changed this: `} else if (currentLiveDocs == null) {`
to this: `} else if (liveDocs == null) {`

f. Changed this: `this.numDocs = bs.cardinality();`
to this: `numDocs = roleQueryBits.cardinality();`

g. Changed this: `int localNumDocs = 0;`
to this: `int counter = 0;`

h. Changed this: `DocIdSetIterator it = new BitSetIterator(bs, 0L);`
to this: `DocIdSetIterator iterator = new BitSetIterator(roleQueryBits, 0L);`

i. Changed this: `for (int doc = it.nextDoc(); doc !=`
to this: `for (int docId = iterator.nextDoc(); docId !=`

j. Changed this: `DocIdSetIterator.NO_MORE_DOCS; doc = it.nextDoc()) {`
to this: `DocIdSetIterator.NO_MORE_DOCS; docId = iterator.nextDoc()) {`

k. Changed this: `if (currentLiveDocs.get(doc)) {`
to this: `if (liveDocs.get(docId)) {`

l. Changed this: `localNumDocs++;`
to this: `counter++;`

m. Changed this: this.numDocs = localNumDocs;
to this: numDocs = counter;

n. Changed this: return this.numDocs;
to this: return numDocs;

o. Changed this: return this.numDocs; // cached
to this: return numDocs;

In short, the code on the "floragunn" side of the side-by-side comparison in paragraph 33 doesn't actually exist.

Paragraph 34

Allegation: floragunn's June 7, 2018, changes also included several other alterations to Search Guard that mimic X-Pack, including, at least: (1) changing the computation of Search Guard's BitSet from an inferior IndexSearcher to align itself with how X-Pack computes the BitSet; and (2) changing computation of live documents to match the X-Pack implementation.

Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 34 of the amended complaint.

floragunn further specifically denies that it made changes to "mimic" X-Pack.

Paragraph 35

Allegation: floragunn took efforts to keep its misconduct concealed. For example, the only explanation floragunn provided for the changes it made on June 7 was "Improve dls/fls." This is a strikingly brief explanation in light of the significant changes floragunn had committed to its code base. And such minimal explanation is inconsistent not only with standard computer programming practices but is also inconsistent with floragunn's explanations accompanying its commits of other code.

Response: floragunn denies the allegations in paragraph 35 and likewise denies that it engaged in any misconduct or kept any changes to its code "concealed." floragunn further denies that the comment accompanying its commit is inconsistent with its past practices, which have often included commits with substantial changes and minimal comments, such as:

Repository	Commit ID	Changes	Comment
searchguard	c04d1cc71d5bbb6d7045fc519fcaee8278d1e8a2	603 additions and 210 deletions	"Fix CCS dnfof"
searchguard	dba296587fb1d828d70295478dbf0736492149ed	30 additions and 1 deletion	"Fix ITT-1386"
searchguard	4b5cb3fa077740923863e3a24c1de5fd91996c85	3 additions and 2 deletions	"fix ITT-1383"
searchguard-enterprise-modules	62b094177484c544a22134fcaaf8fce1b5e5e34c	299 additions and 4 deletions.	"Make LDAP work with Java 11"
searchguard-enterprise-modules	2b5305cc4de75a814e00a21ccc95683171035552	411 additions and 12 deletions	Fix ITT-1563
searchguard-enterprise-modules	bfda8511bc0a360ed7a967b787dd349a889986f4	1,861 additions and 222 deletions	Landed SAML support
searchguard-enterprise-modules	a2bc973f890da4f3b06e7a64b6a4f4318dbeaaa2	224 additions and 13 deletions	Fix ITT-1269 and ITT-1268
searchguard-enterprise-modules	8be1c2c3a12114c1659e9111a13fc792a276a4e2	46 additions and 52 deletions.	fix dls perf regression
searchguard-enterprise-modules	08cb5d512b353d94b5214ce958bd2c94d7906415	140 additions and 101 deletions.	Fix ITT-1245 and ITT-1244
searchguard-enterprise-modules	14daed88dbd0de01d90c0dc20de85cefb87edc9b	324 additions and 53 deletions	bug fixes, more test cases
searchguard-enterprise-modules	700c1e1fa84f754c2855416bf5137d445dc50aa2	106 additions and 53 deletions	json patch diffs
searchguard-enterprise-modules	379cdab523f07563db56afeac7826ad4f02c4323	100 additions and 520 deletions.	Fix mt interceptor
search-guard-rest-api	d0232a4b11b88d62ed8317e558c8341a5f8b0d38	139 additions and 11 deletions	added roles
new-proxy-module	f314ac612b43d916d8d2534923d06d3a2f2987ce	158 additions and 137 deletions	Bug fixes & renaming

Paragraph 36

> Allegation: floragunn's June 7, 2018, changes also lack evidence that floragunn undertook unit testing of the code—yet another absence that is inconsistent with common programming practice and different from floragunn's other public code. This too strongly suggests that floragunn simply copied Elastic's code.

Response: floragunn denies the allegations in paragraph 36 of the amended complaint. floragunn specifically denies that new unit tests were required by the commit, which did not add new functionality to the code, but rather addressed execution speed and performance. floragunn

denies that the commit was inconsistent with common programming practice, and denies that it was different from its past practice. floragunn further denies that anything "suggests" copying.

Paragraph 37

Allegation: Examination of floragunn's Search Guard code reveals that its recent acts of infringement are consistent with a larger and longstanding pattern of misconduct.

Response: floragunn denies the allegations set out in paragraph 37 of the amended complaint. floragunn denies that there are any "acts of infringement" and denies that there was any "misconduct."

Paragraph 38

Allegation: Code released by floragunn as part of Search Guard in 2016 contains the following commented out—that is, non-functional—code:

```
// "internal:*",
// "indices:monitor/*",
// "cluster:monitor/*",
// "cluster:admin/reroute",
// "indices:admin/mapping/put"
```

Response: floragunn admits the allegations in paragraph 38, but specifically denies that these comments are due to any copying of Elastic code. Instead, these five lines document the "action names" generated by Elasticsearch while it runs.

Paragraph 39

Allegation: That code was copied verbatim from the following functional Elastic code in Shield (Elastic's security product that preceded X-Pack) that was released in or before 2015:

```
        protected static final Predicate<String> PREDICATE =
new AutomatonPredicate(patterns(
        "internal:*",
        "indices:monitor/*", // added for marvel
        "cluster:monitor/*", // added for marvel
        "cluster:admin/reroute", // added for DiskThresholdDecider.DiskListener
        "indices:admin/mapping/put" // ES 2.0
MappingUpdatedAction -
updateMappingOnMasterSynchronously
        ));
```

Response: floragunn denies the allegations in paragraph 39, and incorporates by reference its response to paragraph 38.

Paragraph 40

> Allegation: Elastic had not publicly released this source code for Shield at the time of floragunn's copying and/or creation of derivative works from that code. Elastic is informed and believes and, on that basis, alleges that floragunn decompiled Elastic's binaries or otherwise gained access to Elastic's source code to create the copies and/or derivative works referenced in Paragraph 38.

Response: floragunn denies that it copied or created derivative works, and on that basis denies the first sentence. floragunn denies that it decompiled Elastic's binaries, or that it otherwise gained access to Elastic's source code to copy the lines referenced in paragraph 38, and on that basis denies the second sentence. floragunn further incorporates by reference its response to paragraph 38.

Paragraph 41

> Allegation: Code released by floragunn on June 6, 2016, into the search-guard-module-dlsfls repository for Search Guard contains the following:

```
@Override
public void binaryField(final FieldInfo, final byte[] value) throws IOException {

if (fieldInfo.name.equals("_source")) {
            final BytesReference bytesRef = new BytesArray(value);
 final Tuple<XContentType, Map<String, Object>> bytesRefTuple =
 XContentHelper.convertToMap(bytesRef, false);
 final Map<String, Object> filteredSource =
 XContentMapValues.filter(bytesRefTuple.v2(), includes, null);
 final XContentBuilder xBuilder =
 XContentBuilder.builder(bytesRefTuple.v1().xContent()).map(filteredSource);
          delegate.binaryField(fieldInfo, xBuilder.bytes().toBytes());
} else {
   delegate.binaryField(fieldInfo, value);
       }
}
```

Response: floragunn admits the allegations in paragraph 41, but specifically denies that the code is the result of copying of Elastic code.

Paragraph 42

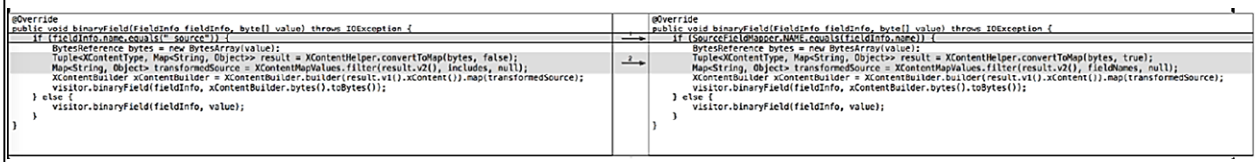
Allegation: That code is substantively identical to the following Elastic code that had previously been included in Shield:

```
@Override
public void binaryField(FieldInfo, byte[] value) throws IOException {
if (SourceFieldMapper.NAME.equals(fieldInfo.name)) {
// for _source, parse, filter out the fields we care about, and serialize back
downstream
BytesReference bytes = new BytesArray(value);
        Tuple<XContentType, Map<String, Object>> result =
XContentHelper.convertToMap(bytes, true);
Map<String, Object> transformedSource = XContentMapValues.filter(result.v2(),
fieldNames, null);
    XContentBuilder =
XContentBuilder.builder(result.v1().xContent()).map(transformedSource);
visitor.binaryField(fieldInfo, xContentBuilder.bytes().toBytes());
    } else {
visitor.binaryField(fieldInfo, value);
    }
    }
```

Response: floragunn denies the allegations in paragraph 42. floragunn further denies that its code is "substantively identical" to Elastic's code that had previously been included in Shield. floragunn specifically denies that any similarities are due to copying of any protectable expression original to Elastic.

Paragraph 43

Allegation: Ignoring non-substantive differences in the code, it is clear that the floragunn code (on the left) is copied from or, at least, a derivative work of the Elastic code (on the right). A larger version of this graphic is attached to this Complaint as Exhibit C.



Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 43 of the amended complaint. floragunn further denies that the changes Elastic proposes in paragraph 43 render anything "clear." floragunn specifically

denies that the side-by-side comparison in paragraph 43, which Elastic created by deleting code text that appears in floragunn's code and inserting code text that does not appear in floragunn's code, as well as re-formatting floragunn's source code text, supports the conclusions Elastic purports to make. floragunn notes that Elastic's changes to floragunn's actual code include:

a. Changed this:

```
public void binaryField(final FieldInfo, final byte[] value) throws  IOException {
```

to this:

```
public void binaryField(FieldInfo fieldInfo, byte[] value) throws IOException {
```

b. Changed this: `final BytesReference bytesRef = new BytesArray(value);`
to this: `BytesReference bytes = new BytesArray(value);`

c. Changed this: `final Tuple<XContentType, Map<String, Object>> bytesRefTuple=`
to this: `Tuple<XContentType, Map<String, Object>> result =`

d. Changed this: `XContentHelper.convertToMap(bytesRef, false);`
to this: `XContentHelper.convertToMap(bytes, false);`

e. Changed this: `final Map<String, Object> filteredSource =`
to this: `Map<String, Object> transformedSource =`

f. Changed this: `XContentMapValues.filter(bytesRefTuple.v2(), includes, null);`
to this: `XContentMapValues.filter(result.v2(), includes, null);`

g. Changed this: `final XContentBuilder xBuilder =`
to this: `XContentBuilder xContentBuilder =`

h. Changed this:

```
XContentBuilder.builder(bytesRefTuple.v1().xContent()).
map(filteredSource);
```

to this:

```
XContentBuilder.builder(result.v1().xContent()).
map(transformedSource);
```

i. Changed this: `delegate.binaryField(fieldInfo, xBuilder.bytes().toBytes());`
to this: `visitor.binaryField(fieldInfo, xContentBuilder.bytes().toBytes());`

j. Changed this: `delegate.binaryField(fieldInfo, value);`
to this: `visitor.binaryField(fieldInfo, value);`

Paragraph 44

Allegation: Elastic had not publicly released this source code for Shield at the time of floragunn's copying and/or creation of derivative works from that code. Elastic is informed and believes and, on that basis, alleges that floragunn decompiled Elastic's binaries or otherwise gained access to Elastic's source code to create the copies and/or derivative works referenced in Paragraph 41.

Response: floragunn denies that it copied or created derivative works, and on that basis denies the first sentence. floragunn denies that it decompiled Elastic's binaries, or that it otherwise gained access to Elastic's source code to copy the material referenced in paragraph 41, and on that basis denies the second sentence.

Paragraph 45

Allegation: Infringement by floragunn is evident in additional code in the ShieldNettyHttpServerTransport file. Code released by floragunn on December 10, 2016 as part of the Search Guard SearchGuardSSLNettyHttpServerTransport file contains the following content:

```
@Override
  protected void exceptionCaught(ChannelHandlerContext ctx, ExceptionEvent e) throws
  Exception {
     if(this.lifecycle.started()) {

        final Throwable cause = e.getCause();
        if(cause instanceof NotSslRecordException) {
            logger.warn("Someone speaks plaintext instead of ssl, will close the channel");
            ctx.getChannel().close();
            return;
        } else if (cause instanceof SSLException) {
            logger.error("SSL Problem "+cause.getMessage(),cause);
            ctx.getChannel().close();
            return;
                  } else if (cause instanceof SSLHandshakeException) {
            logger.error("Problem during handshake "+cause.getMessage());
            ctx.getChannel().close();
            return;
        }
     }

     super.exceptionCaught(ctx, e);
  }
```

Response: floragunn denies that it has engaged in any copyright infringement, admits that floragunn released the code in paragraph 45, but denies that the code was released on December 10, 2016.

Paragraph 46

Allegation: That code is substantively identical to the following Elastic code included in the binary of Elastic Shield released June 24, 2015:

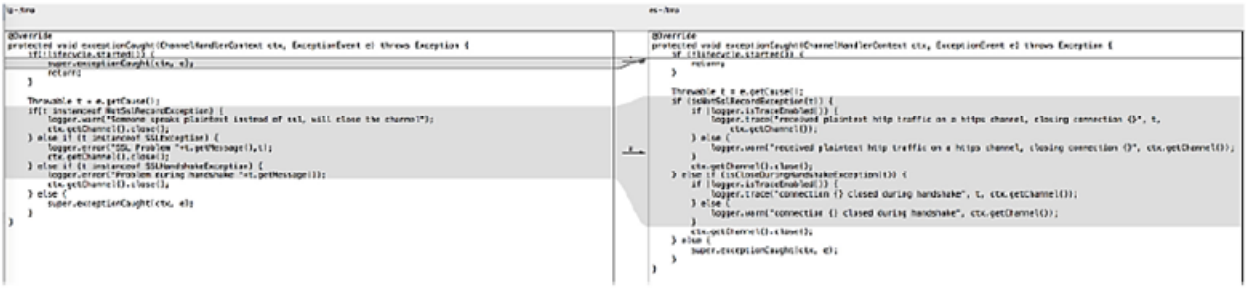
```
@Override
protected void exceptionCaught(ChannelHandlerContext ctx, ExceptionEvent e)
throws
Exception {
if (!lifecycle.started()) {
return;
}
Throwable t = e.getCause();
if (isNotSslRecordException(t)) {
if (logger.isTraceEnabled()) {
logger.trace("received plaintext http traffic on a https channel, closing
connection
{}", t, ctx.getChannel());
} else {
logger.warn("received plaintext http traffic on a https channel, closing
connection
{}", ctx.getChannel());
}
ctx.getChannel().close();
} else if (isCloseDuringHandshakeException(t)) {
if (logger.isTraceEnabled()) {
logger.trace("connection {} closed during handshake", t, ctx.getChannel());
} else {
logger.warn("connection {} closed during handshake", ctx.getChannel());
}
ctx.getChannel().close();
} else {
super.exceptionCaught(ctx, e);
}
}
```

Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 46. floragunn further denies that its code is "substantively identical" to code in the binary of Elastic Shield released on June 25, 2015.

floragunn specifically denies that any similarities are due to copying of any protectable expression original to Elastic.

Paragraph 47

Allegation: Ignoring non-substantive differences in the code, it is clear that the floragunn code (on the left) is copied from or, at least, a derivative work of the Elastic code (on the right). A larger version of this graphic is attached to this Complaint as Exhibit D.



Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 47 of the amended complaint. floragunn further denies that the changes Elastic proposes in paragraph 47 render anything "clear." floragunn specifically denies that the side-by-side comparison in paragraph 47, which Elastic created by deleting code text that appears in floragunn's code and inserting code text that does not appear in floragunn's code, as well as re-formatting floragunn's source code text, supports the conclusions Elastic purports to make. floragunn notes that Elastic's changes to floragunn's actual code include:

a. Changed this: `if(this.lifecycle.started()) {`
to this: `if(!lifecycle.started()) {`

b. Added this:
```
super.exceptionCaught(ctx, e);
return;
}
```

c. Changed this: `final Throwable cause = e.getCause();`
to this: `Throwable t = e.getCause();`

d. Changed this: `if(cause instanceof NotSslRecordException) {`
to this: `if(t instanceof NotSslRecordException) {`

e. Changed this:
```
                return;
            } else if (cause instanceof SSLException) {
```
to this:
```
            } else if (t instanceof SSLException) {
```

f. Changed this:
```
logger.error("SSL Problem "+cause.getMessage(),cause);
```
to this:
```
logger.error("SSL Problem "+t.getMessage(),t);
```

g. Changed this:
```
                return;
            } else if (cause instanceof SSLHandshakeException) {
```
to this:
```
            } else if (t instanceof SSLHandshakeException) {
```

h. Changed this:
```
logger.error("Problem during handshake "+cause.getMessage());
```
to this:
```
logger.error("Problem during handshake "+t.getMessage());
```

i. Changed this:
```
                return;
            }
        }
        super.exceptionCaught(ctx, e);
    }
```

to this:
```
    } else {
        super.exceptionCaught(ctx, e);
    }
}
```

Paragraph 48

Allegation: Elastic had not publicly released this source code for Shield at the time of floragunn's copying and/or creation of derivative works from that code. Elastic is informed and believes and, on that basis, alleges that floragunn decompiled Elastic's binaries or otherwise gained access to Elastic's source code to create the copies and/or derivative works referenced in Paragraph 45.

Response: floragunn denies that it copied or created derivative works, and on that basis denies the first sentence. floragunn denies that it decompiled Elastic's binaries, or that it otherwise gained access to Elastic's source code to copy the material referenced in paragraph 45, and on that basis denies the second sentence.

Paragraph 49

Allegation: Subsequent investigation has also revealed floragunn's copying and/or creation of derivative works from code from the X-Pack plugin for Elastic's Kibana product. The infringed code that Elastic has identified comes from the X-Pack Kibana elements Get Next URL, Saved Objects Client, AngularJS Management Screens, callWithRequestFactory, and fetchAllFromScroll. In addition to the examples below, Elastic has identified copying and/or creation of derivative works in April 5, 2017, August 6, 2017, and June 4, 2019 commits to Search Guard.

Response: floragunn denies that it has copied or created derivative works from code from the X-Pack plugin for Elastic's Kibana product, and therefore denies the allegations of paragraph 49 of the amended complaint.

Paragraph 50:

Allegation: As one example, Search Guard code released by floragunn on March 31, 2018 in get_next_url.js contains the following code:

```
const {query, hash} = parse(currentUrl, true);
if (!query.nextUrl) {
    return `${basePath}/`;
}
const { protocol, hostname, port, pathname } = parse(query.nextUrl);
if (protocol || hostname || port) {
    return `${basePath}/`;
}
if (!String(pathname).startsWith(basePath)) {
    return `${basePath}/`;
}
return query.nextUrl + (hash || '');
```

Response: floragunn admits that the Search Guard code released on March 31, 2018 contained that code, but denies that the code is "one example" of anything.

Paragraph 51:

Allegation: That code closely mirrors the following code that Elastic included in a bug fix to the X-Pack Kibana plugin in parse_next.js on April 4, 2017:

```
const { query, hash } = parse(href, true);
if (!query.next) {
  return `${basePath}/`;
}
const { protocol, hostname, port, pathname } = parse(query.next);
if (protocol || hostname || port) {
  return `${basePath}/`;
}
if (!String(pathname).startsWith(basePath)) {
  return `${basePath}/`;
}
return query.next + (hash || '');
```

Response: floragunn denies that it has engaged in any copyright infringement and

therefore denies Elastic's allegations in paragraph 51 of the amended complaint. floragunn further denies its code "closely mirrors" Elastic's code. floragunn specifically denies that any similarities are due to copying of any protectable expression original to Elastic.

Paragraph 52:

> Allegation: Ignoring non-substantive differences in the code, it is clear that the floragunn code in Paragraph 50 is copied from, or at least a derivative work of, the Elastic code in Paragraph 51.

Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 52 of the amended complaint. floragunn denies that it is "clear that" the floragunn code in paragraph 50 is copied from or at least a derivative work of, the Elastic code in paragraph 51, and specifically denies that any similarities are due to copying of any protectable expression original to Elastic.

Paragraph 53:

> Allegation: As another example, Search Guard code released by floragunn on October 28, 2018 in get_next_url.js contains the following code:

```
const {query, hash} = parse(currentUrl, true, true);
if (!query.nextUrl) {
  return `${basePath}/`;
}
const { protocol, hostname, port, pathname } = parse(query.nextUrl, false, true);
if (protocol !== null || hostname !== null || port !== null) {
  return `${basePath}/`;
}
if (!String(pathname).startsWith(basePath)) {
  return `${basePath}/`;
}
return query.nextUrl + (hash || '');
```

Response: floragunn admits that the Search Guard code released on October 28, 2018 contained that code, but denies that the code is "another example" of anything.

Paragraph 54:

Allegation: That code is nearly identical to the following code that Elastic included in a bug fix to the X-Pack Kibana plugin in parse_next.js on January 28, 2018:

```
const {query, hash} = parse(href, true);
if (!query.next) {
  return `${basePath}/`;
}
const { protocol, hostname, port, pathname } = parse(
  query.next,
  false,
  true
);
if (protocol !== null || hostname !== null || port !== null) {
  return `${basePath}/`;
}
if (!String(pathname).startsWith(basePath)) {
  return `${basePath}/`;
}
return query.next + (hash || '');
 }
```

Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 54 of the amended complaint. floragunn further denies its code is "nearly identical" to Elastic's code. floragunn specifically denies that any similarities are due to copying of any protectable expression original to Elastic.

Paragraph 55:

Allegation: Ignoring non-substantive differences in the code, it is clear that the floragunn code in Paragraph 53 is copied from, or at least a derivative work of, the Elastic code in Paragraph 54.

Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 55 of the amended complaint. floragunn denies that it is "clear" that the floragunn code in paragraph 53 is copied from, or at least a derivative work of, the Elastic code in paragraph 54.

Paragraph 56:

Allegation: As another example, Search Guard code released by floragunn on October 28, 2018 in get_next_url.js contains the following code:

```
import { once } from 'lodash';
import { elasticsearchSignalsPlugin } from '../elasticsearch_signals_plugin';
import { CLUSTER } from '../../../../utils/signals/constants';
const callWithRequest = once((server) => {
  const { callWithRequest } =
server.plugins.elasticsearch.createCluster(CLUSTER.ALERTING, {
    plugins: [elasticsearchSignalsPlugin]
  });
  return callWithRequest;
});
export const callWithRequestFactory = (server, request) =>
  (...rest) => callWithRequest(server)(request, ...rest)
```

Response: floragunn admits that the Search Guard code released on October 28, 2018 contained that code, but denies that the code is "another example" of anything.

Paragraph 57:

Allegation: That code is nearly identical to the following Elastic code that occurs multiple places within the X-Pack Kibana plugin, including in a February 28, 2019 commit to call_with_request_factory.js reproduced here:

```
import { once } from 'lodash';
import { elasticsearchJsPlugin } from '../elasticsearch_js_plugin';
const callWithRequest = once((server) => {
  const config = { plugins: [ elasticsearchJsPlugin ] };
  const cluster = server.plugins.elasticsearch.createCluster('watcher', config);

  return cluster.callWithRequest;
});
export const callWithRequestFactory = (server, request) => {
  return (...args) => {
    return callWithRequest(server)(request, ...args);
  };
};
```

Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 57 of the amended complaint. floragunn further denies its code is "nearly identical" to Elastic's code. floragunn specifically denies that any similarities are due to copying of any protectable expression original to Elastic.

Paragraph 58:

Allegation: Ignoring non-substantive differences in the code, it is clear that the floragunn code in Paragraph 56 is copied from, or at least a derivative work of, the Elastic code in Paragraph 57.

Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 58 of the amended complaint. floragunn denies that it is "clear that" the floragunn code in paragraph 56 is copied from or at least a derivative work of, the Elastic code in paragraph 57, and specifically denies that any similarities are due to copying of any protectable expression original to Elastic.

Paragraph 59:

Allegation: As one more example, Search Guard code released by floragunn also on August 30, 2019 in fetch_all_from_scroll.js contains the following code:

```
import { ES_SCROLL_SETTINGS } from '../../../../utils/signals/constants';
export function fetchAllFromScroll(response, callWithRequest, allHits = []) {
  const { _scroll_id: scrollId, hits: { hits = [] } } = response;
  if (hits.length) {
    allHits.push(...hits);
    return callWithRequest('scroll', {
      body: {
        scroll: ES_SCROLL_SETTINGS.KEEPALIVE,
        scroll_id: scrollId
      }
    }).then(_response => fetchAllFromScroll(_response, callWithRequest, allHits));
return Promise.resolve(allHits);
}
```

Response: floragunn admits that the Search Guard code released on August 30, 2019 contained that code, but denies that the code is "one more example" of anything.

Paragraph 60:

Allegation: That code very closely mirrors the following Elastic code included in an April 6, 2017 commit to fetch_all_from_scroll.js reproduced here:

```
import { get } from 'lodash';
import { ES_SCROLL_SETTINGS } from '../../../common/constants';
export function fetchAllFromScroll(response, callWithRequest, hits = []) {
  const newHits = get(response, 'hits.hits', []);
  const scrollId = get(response, '_scroll_id');
  if (newHits.length > 0) {
    hits.push(...newHits);
    return callWithRequest('scroll', {
      body: {
        scroll: ES_SCROLL_SETTINGS.KEEPALIVE,
        scroll_id: scrollId
      }
    })
      .then(innerResponse => {
        return fetchAllFromScroll(innerResponse, callWithRequest, hits);
      });
  }
  return Promise.resolve(hits);
}
```

Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 60 of the amended complaint. floragunn further denies its code "very closely mirrors" Elastic's code. floragunn specifically denies that any similarities are due to copying of any protectable expression original to Elastic.

Paragraph 61:

> Allegation: Ignoring non-substantive differences in the code, it is clear that the floragunn code in Paragraph 59 is copied from, or at least a derivative work of, the Elastic code in Paragraph 60.

. Response: floragunn denies that it has engaged in any copyright infringement and therefore denies Elastic's allegations in paragraph 61 of the amended complaint. floragunn denies that it is "clear" that the floragunn code in paragraph 59 is copied from, or at least a derivative work of, the Elastic code in paragraph 60, and specifically denies that any similarities are due to copying of any protectable expression original to Elastic

Paragraph 62:

> Allegation: A comment made by a floragunn programmer in a June 4, 2019 Search Guard commit provides further proof of copying because it indicates that the programmer did not understand the reason that a variable in the source code was formatted in a particular way. Programmers must choose the format or formats for the names of variables in their source code. This choice is often more than stylistic, because it can affect compatibility with other programs and operating systems. One such format is "snake case" where a programmer replaces all spaces with "_." Accordingly, a variable named "first variable" would be written in snake case as "first_variable."

Response: floragunn denies that the comment provides "further proof of copying" and therefore denies allegations in the first sentence of paragraph 62. floragunn admits the allegations in the remaining sentences of paragraph 62.

Paragraph 63:

> Allegation: Elastic's X-Pack Kibana plugin formats variables in "snake case" to remain compatible with the X-Pack plugin for Elasticsearch—but, on information and belief, that reason for formatting variables in "snake case" is not present for the infringing Search Guard code. Although the reason for use of "snake case" is absent, floragunn's infringing code also uses "snake case" for the "bulk_create" variable. But a floragunn programmer left a comment in the infringing Search Guard code noting that s/he could not determine why the code used "snake case" for the "bulk_create" variable, writing: "@todo Why the snake case here? What do our permissions look like."

Response: floragunn denies having sufficient information to respond to the allegations in the first sentence of paragraph 63 regarding Elastic's X-Pack variables, and therefore denies such allegations. floragunn denies that it has "infringing code" and on that basis denies allegations in the remainder of the first sentence of paragraph 63. floragunn denies that it has "infringing code" and on that basis denies allegations in the second sentence of paragraph 63. floragunn admits that the comment alleged in the third sentence of paragraph 63 was in its code, but denies the remainder of the allegations in the third sentence of paragraph 63.

Paragraph 64

> Allegation: floragunn's Search Guard product directly competes with the security features in Elastic's X-Pack and X-Pack Kibana plugin.

Response: floragunn admits the allegations in paragraph 64 of the amended complaint.

Paragraph 65

> Allegation: Elastic is informed and believes, and, on that basis alleges that floragunn knew that Elastic had its principal place of business in the Northern District of California.

Response: floragunn admits the allegations in paragraph 65 of the amended complaint, except that it is floragunn's understanding that elasticsearch B.V.'s principal place of business is in the Netherlands.

Paragraph 66

> Allegation: floragunn maintains significant and ongoing commercial ties to the Northern District of California. The industry that provides security features for Elastic Stack is very small, and, Elastic is informed and believes, is composed of at most six companies. Despite the small number of companies providing security features for Elastic Stack, the customer base for Elastic Stack security features is broad. floragunn boasts of a "global customer base," including "many of the tech giants." Due to the prominence of the technology industry in the Northern District of California, many of these companies are headquartered in, maintain offices in, or do significant business in the Northern District of California.

Response: floragunn denies having sufficient information to respond to the allegations in paragraph 66 of the amended complaint, and therefore denies such allegations.

Paragraph 67

> Allegation: Before Elastic commenced this lawsuit, floragunn hosted its infringing source code on a website run by GitHub, Inc. GitHub, Inc. is headquartered and maintains its principal place of business in San Francisco, California, within the Northern District of California. floragunn currently hosts infringing source code through GitLab Inc., a company also headquartered in San Francisco, California, within the Northern District of California.

Response: floragunn denies the allegations in paragraph 67 of the amended complaint, because as explained in floragunn's answers to Elastic's allegations above, floragunn

has never hosted "infringing code." In addition, floragunn does not host its source code through GitLab, Inc.

Paragraph 68

> Allegation: Further, Elastic is informed and believes, and, on that basis, alleges that, floragunn made commercial use of its infringing Search Guard product by purposefully marketing and licensing that product to customers in the Northern District of California. By way of example, Elastic is informed and believes, and, on that basis, alleges that floragunn licensed its Search Guard software to: (1) PayPal Holdings, Inc., a company that, on information and belief, has its principal place of business in San Jose, California; (2) AppsCode, a company that, on information and belief, has its principal place of business in San Leandro, California, for use in AppsCode's CubeDB software; (3) NVIDIA, a company that, on information and belief, has its principal place of business in Santa Clara, California; (4) Zuora, a company that, on information and belief, has its principal place of business in San Mateo, California; and (5) OpenTable, Inc., a company that, on information and belief, has its principal place of business in San Francisco, California.

Response: floragunn denies the allegations in paragraph 68 of the amended complaint, because as explained in floragunn's answers to Elastic's allegations above, Search Guard is not an "infringing product."

Paragraph 69

> Allegation: Additionally, over a span of several days in March 2019, floragunn actively promoted Search Guard to California entities and individuals while hosting a booth at a data security conference at the Moscone Center in San Francisco, California.

Response: floragunn admits that it hosted a booth at a data security conference in San Francisco in March 2019 to promote Search Guard. floragunn denies having knowledge and information as to whether it promoted Search Guard to "California entities and individuals," and therefore denies such allegations.

Paragraph 70

> Allegation: floragunn's marketing and commercial licensing of a directly competing product that infringes Elastic's copyright demonstrates an intent

knowingly to harm Elasticsearch, Inc. a company with its principal place of business in Mountain View, California. It further shows that floragunn directed its infringing activities at the Northern District of California.

Response: floragunn denies the allegations in paragraph 70 of the amended complaint because it has not engaged in any "infringing activities" in this District or any other.

Paragraph 71

Allegation: floragunn's infringement of Elastic's copyright has caused and continues to cause Elastic injury in the Northern District of California.

Response: floragunn denies the allegations in paragraph 71 of the amended complaint, because as explained in floragunn's answers to Elastic's allegations above, floragunn has not infringed any of Elastic's copyrights.

Paragraph 72

Allegation: floragunn's marketing and distribution of infringing Search Guard software causes third party Search Guard users to incorporate code that infringes Elastic's copyrights in X-Pack and the X-Pack Kibana plugin. Those third parties therefore necessarily reproduce and use Elastic's proprietary X-Pack and/or X-Pack Kibana plugin code when they incorporate Search Guard into their adoptions of Elasticsearch, thereby infringing Elastic's copyrights.

Response: floragunn denies the allegations in paragraph 72 of the amended complaint, because as explained in floragunn's answers to Elastic's allegations above, floragunn has not infringed any of Elastic's copyrights.

Paragraph 73

Allegation: Additional third parties have incorporated floragunn's infringing code into products and services they offer publicly. Elastic has investigated to identify third parties who have incorporated floragunn's infringing code into their products and services.

Response: floragunn denies the allegations in paragraph 73 of the amended complaint, because as explained in floragunn's answers to Elastic's allegations above, floragunn has not infringed any of Elastic's copyrights.

Paragraph 74

> Allegation: Among other infringing third party products and services that Elastic has identified, Amazon.com, Inc.'s and Amazon Web Services Inc.'s Open Distro for Elasticsearch ("Open Distro") and Amazon Elasticsearch Service ("AESS") offerings both contain and/or contained infringing code that originated with floragunn. Open Distro contains and/or contained infringing code related to floragunn's infringement of Elastic's copyrights in X-Pack and the X-Pack Kibana plugin. AESS contains or contained infringing code related to floragunn's infringement of Elastic's copyrights in X-Pack. Rackspace US, Inc.'s ObjectRocket for Elasticsearch contains or contained infringing code related to infringement of Elastic's copyrights in X-Pack and the X-Pack Kibana plugin. And IBM Corporation's IBM Cloud Databases for Elasticsearch contains or contained infringing code related to infringement of Elastic's copyrights in X-Pack.

Response: floragunn denies the allegations in paragraph 74 of the amended complaint, because as explained in floragunn's answers to Elastic's allegations above, floragunn has not infringed any of Elastic's copyrights, and therefore users of Search Guard do not use infringing code.

Paragraph 75

> Allegation: floragunn is undoubtedly aware that its conduct is unlawful. On the website for Search Guard, floragunn states that, just because "the source code of a piece of software is available for anyone to view and inspect," that "does not necessarily mean that the product is available at no cost, and it does not mean that it is solely a community product." floragunn goes on to warn "it is illegal to take our enterprise features into production without purchasing a license. ***This can lead to serious legal consequences, which can bring more harm and costs to a company*** . . . ." (emphasis added).

Response: floragunn denies the allegations in paragraph 75 of the amended complaint because it has done nothing "unlawful," and further respectfully directs the Court to floragunn's response to paragraph 76 of the amended complaint, below.

Paragraph 76

> Allegation: floragunn actively sought to avoid United States copyright law. On September 4, 2019, Elastic submitted Notices of Copyright Infringements under the DMCA to two websites that hosted floragunn's infringing code, GitHub, Inc.

> and Sonatype Inc. On or about September 11, 2019, GitHub and Sonatype removed floragunn's infringing code from their websites. Pursuant to the DMCA, floragunn then had the opportunity to submit counter notifications stating that Elastic's assertion that their content infringed Elastic's copyrights was mistaken. Submitting a counter notification could potentially have led to GitHub and Sonatype restoring floragunn's content to their websites. Elastic is not aware of any such counter notification by floragunn, and, on information and belief, floragunn did not take this opportunity to assert that its code did not infringe Elastic's copyright.

Response: floragunn denies the allegations in paragraph 76 of the amended complaint and further states that the allegations in the entire section of Elastic's amended complaint titled "Floragunn Attempts to Avoid Enforcement of United States Copyright Law" (paragraphs 75-80) are gratuitous attempts to smear floragunn. Elastic makes no legal claims in its pleadings against floragunn based any of floragunn's alleged actions related to DMCA. For example, Elastic's statement that "floragunn actively sought to avoid United States copyright law" is false. Nothing floragunn has done in response to the DMCA notice can reasonably be described as "avoiding" the law. Elastic mischaracterizes what the DMCA process is, and why floragunn moved its source code to an offshore host after GitHub took down floragunn's content after being served with a DMCA notice. Likewise, Elastic's claim that it had an opportunity to submit counter notifications omits that the DMCA provides that a counter notification will have no effect if "the person who submitted the notification . . . has filed an action . . . ." 17 U.S.C. § 512(g)(2)(B). Because Elastic filed its original complaint in this action on September 4, 2019, a counter notification would have had no effect. On September 12, 2019, floragunn moved its source code to a repository hosted by AWS (Amazon). Elastic followed up immediately with another DMCA notice to Amazon, which resulted in floragunn voluntarily removing the content from the AWS site. At that point, floragunn's only practical recourse to protect its code from being wrongfully taken down by Elastic was to use a host that would not seek to avail itself of the safe harbor protection of DMCA.

Paragraph 77

> Allegation: What floragunn did do, however, was switch the hosting of its infringing content to a different provider: Amazon Web Services, Inc. Amazon Web Services, Inc. maintains multiple offices in the Northern District of California.

Response: floragunn denies the allegations in paragraph 77 of the amended complaint because it has never sought to host "infringing content," and respectfully directs the Court to floragunn's response to Paragraph 76 above.

Paragraph 78

> Allegation: On September 12, 2019, Elastic sent a Notice of Copyright Infringements under the DMCA to Amazon Web Services, Inc., and Amazon Web Services, Inc. removed floragunn's infringing code from its website on September 13, 2019. On information and belief, floragunn again did not take the opportunity to submit a counter notification and assert that its content did not infringe Elastic's copyrights. Once again, however, floragunn switched the hosting of its infringing content to a different provider.

Response: floragunn denies the allegations in paragraph 78 of the amended complaint because it has never sought to host "infringing code," and respectfully directs the Court to floragunn's response to Paragraph 76 above.

Paragraph 79

> Allegation: It appears that floragunn's choice of the new host for its downloads was driven by a desire by floragunn to avoid takedowns required by the law of the United States. floragunn began hosting its infringing downloads through BlueAngelHost PVT. LTD. BlueAngelHost PVT. LTD. advertises "DMCA Ignored Hosting." It boasts that "***Purchasing USA-based hosting for a site that is not legal to be run in America is not a sensible thing to do. Offshore hosting can be helpful for less scrupulous businesses who wish to bypass local laws or regulations, particularly for issues like copyright law, which is also known as no DMCA hosting***" (emphasis added). BlueAngelHost PVT. LTD. lists a postal address in Serbia on its website and advertises data centers in Bulgaria, Russia, and the Netherlands.

Response: floragunn denies the allegations in paragraph 79 of the amended complaint because it has never sought to host "infringing code," or hosted "infringing downloads" and because it is false that DMCA "takedowns are required by the law of the United States." Elastic's statement in paragraph that 79 of the amended complaint that DMCA "takedowns [are] required by the law of the United States" is false; instead, the DMCA provides certain safe harbor protections for hosts that take down material in response to a DMCA notice, *even if* the claim of infringement is incorrect. Likewise, floragunn's decision to use a host that would not remove content simply based on an *accusation* of infringement is not an attempt to "avoid" copyright law. floragunn further respectfully directs the Court to floragunn's response to Paragraph 76 above.

Paragraph 80

> Allegation: Months after this lawsuit was filed, floragunn continues to include infringing code in Search Guard. In early October 2019, floragunn released new versions of its Search Guard products through web services run by GitLab Inc., Sonatype Inc., and floragunn's own website. These new versions purport to remove the infringing code that Elastic identified in its initial complaint. However, floragunn did not remove all instances of copying. The new versions of Search Guard continue to, at least, contain code that infringes Elastic's copyrights in its X-Pack Kibana plugin, as identified in this First Amended Complaint. Elastic continues to investigate floragunn's Search Guard code for instances of infringement and may identify further infringement.

Response: floragunn denies the allegations in paragraph 80 of the amended complaint because Search Guard has never contained "infringing code" nor did floragunn remove "all instances of copying," from its code since floragunn never copied anything from X-Pack in the first place. floragunn further notes that Elastic has had four years to "investigate" floragunn's publicly available code for instances of infringement, and has only been able to point to about 100 lines out of more than 60,000 lines of Search Guard code to make its allegations of infringement -- all of which are meritless and rebutted above. Elastic's threat to further "investigate floragunn's

Search Guard code for instances of infringement" speaks more to Elastic's true intentions of harassing and injuring floragunn's business that it does about pursuing meritorious claims of infringement, which do not exist.

Paragraph 81

> Allegation: Elastic incorporates by reference each of the allegations in the preceding paragraphs of this Complaint as if fully set forth here.

Response: floragunn restates and realleges each of its responses in the above paragraphs of this Answer.

Paragraph 82

> Allegation: Before initiating this action, Elastic registered, effective August 14, 2019, versions 1.0.0 and 2.0.0 of Elasticsearch Shield (the predecessor name for X-Pack) and versions 5.0.0, 6.0.0, 6.2.0, 6.2.x, and 6.3.0 of X-Pack under Registration Numbers TX 8-762-996, TX 8-762994, TX 8-762-975, TX 8-762-985, TX 8-762-987, TX 8-762-988, and TX 8-762-991, respectively. Elastic further registered, effective September 9, 2019, versions 1.1.1, 1.3.0, 2.0.0-beta1, and 2.0.0-beta2 of Elasticsearch Shield under Registration Numbers TX 8-773-254, TX 8773-258, TX 8-773-261, and TX 8-773-263, respectively. Elastic additionally registered version 2.3.2 of the Kibana Shield plugin and versions 5.2.0, 5.3.1, 5.6.7, and 6.4.0 of the X-Pack Kibana plugin under Registration Numbers TX 8-796-945, TX 8-777-406, TX 8-777-412, TX 8-778-023, and TX 8-778-024, respectively, effective September 19, 2019; version 5.4.0 of the X-Pack Kibana plugin under Registration Number TX 8-796-010, effective November 4, 2019; and version 7.2.0 of the X-Pack Kibana plugin under Registration Number TX 8-796-013, effective October 31, 2019. Copies of those Certificates of Registration are attached as Exhibits E through V to this First Amended Complaint.

Response: floragunn denies that it has information sufficient to respond to the allegations in paragraph 82 of the amended complaint, and therefore denies such allegations.

Paragraph 83

> Allegation: These works contain copyrightable subject matter for which copyright protection exists under the Copyright Act, 17 U.S.C. § 101, *et seq.* elasticsearch B.V. is the exclusive owner of all rights in these copyrighted works. Elasticsearch, Inc. holds the exclusive license from elasticsearch B.V. to enforce the copyright in and distribute copies of these works in, among other territories, the United States.

Response: floragunn denies that it has information sufficient to respond to the allegations in paragraph 83 of the amended complaint, and therefore denies such allegations.

Paragraph 84

> Allegation: Through the actions described herein, floragunn has infringed and will continue to infringe Elastic's copyrights in the X-Pack and X-Pack Kibana plugin code by, at least, reproducing, preparing derivative works from, and distributing copies of those copyrighted works.

Response: floragunn denies the allegations in paragraph 84 of the amended complaint for the reasons set forth above in this Answer.

Paragraph 85

> Allegation: floragunn's infringing conduct alleged herein was and continues to be willful and with full knowledge of Elastic's rights in the copyrighted works, and that conduct has enabled floragunn to profit illegally from infringement.

Response: floragunn denies the allegations in paragraph 85 of the amended complaint for the reasons set forth above in this Answer.

Paragraph 86

> Allegation: Elastic is entitled to an injunction restraining floragunn, its officers, agents, employees, assigns, and all persons acting in concert with them from engaging in further infringement of Elastic's copyrights.

Response: floragunn denies the allegations in paragraph 86 of the amended complaint for the reasons set forth above in this Answer.

Paragraph 87

> Allegation: Elastic is entitled to recover from floragunn the damages it has sustained and will sustain as a result of floragunn's wrongful acts as alleged herein. Elastic is further entitled to recover from floragunn the gains, profits, and advantages it has obtained as a result of floragunn's wrongful acts. The full extent of Elastic's damages and the gains, profits, and advantages floragunn has obtained by reason of its aforesaid acts of copyright infringement cannot be determined at this time, but will be proven at trial. Further, Elastic is entitled to

recover costs and reasonable attorneys' fees from floragunn as a result of the wrongful acts alleged herein.

Response: floragunn denies the allegations in paragraph 87 of the amended complaint for the reasons set forth above in this Answer.

Paragraph 88

Allegation: Elastic incorporates by reference each of the allegations in the preceding paragraphs of this Complaint as if fully set forth here.

Response: floragunn restates and realleges each of its responses in the above paragraphs of this Answer.

Paragraph 89

Allegation: floragunn's distribution of infringing Search Guard software induces, causes, encourages, and materially contributes to Search Guard users and third parties that incorporate Search Guard code into their products and services infringing Elastic's copyrights in the X-Pack and/or X-Pack Kibana plugin code by engaging in unauthorized reproduction and distribution of works containing Elastic's copyrighted material.

Response: floragunn denies the allegations in paragraph 89 of the amended complaint for the reasons set forth above in this Answer.

Paragraph 90

Allegation: Elastic is informed and believes, and, on that basis, alleges that floragunn derived substantial financial benefit from Search Guard users' and third parties' infringement of Elastic's copyrights in X-Pack and/or the X-Pack Kibana plugin.

Response: floragunn denies the allegations in paragraph 90 of the amended complaint for the reasons set forth above in this Answer.

Paragraph 91

Allegation: floragunn's marketing, commercial distribution of, licensing of, and profit from infringing Search Guard software shows that it knowingly, intentionally, willfully, and purposefully induced, caused, encouraged, and materially contributed to, and continues to knowingly, intentionally, willfully, and purposefully induce, cause, encourage, and materially contributes to, Search

Guard users' and third parties' infringement of Elastic's copyrights in X-Pack and/or the X-Pack Kibana plugin.

Response: floragunn denies the allegations in paragraph 91 of the amended complaint for the reasons set forth above in this Answer.

Paragraph 92

Allegation: floragunn has the ability to prevent Search Guard users and third parties from infringing Elastic's copyrights in the X-Pack and X-Pack Kibana plugin code by omitting the infringing code from its Search Guard software product. However, floragunn has not prevented Search Guard users and third parties from infringing Elastic's copyrights in the X-Pack and X-Pack Kibana plugin code.

Response: floragunn denies the allegations in paragraph 92 of the amended complaint for the reasons set forth above in this Answer, and because Search Guard does not infringe Elastic's code.

Paragraph 93

Allegation: floragunn, through its knowing and intentional inducement, causation, encouragement, and material contribution to the infringement of Elastic's copyrights in the X-Pack and X-Pack Kibana plugin code by Search Guard users and third parties, is committing and/or is contributorily and vicariously liable for the acts of infringement by Search Guard users and third parties. Each act of infringement that floragunn knowingly and intentionally induced, caused, encouraged, and materially contributed to is a separate and distinct act of infringement.

Response: floragunn denies the allegations in paragraph 93 of the amended complaint for the reasons set forth above in this Answer.

Paragraph 94

Allegation: Elastic is entitled to an injunction restraining floragunn, its officers, agents, employees, assigns, and all persons acting in concert with them from actions inducing, causing, encouraging, or materially contributing to Search Guard users' and third parties' infringement of Elastic's copyrights.

Response: floragunn denies the allegations in paragraph 94 of the amended complaint for the reasons set forth above in this Answer.

Paragraph 95

> Allegation: Elastic is entitled to recover from floragunn the damages it has sustained and will sustain as a result of floragunn's acts inducing, causing, encouraging, or materially contributing to Search Guard users' and third parties' infringement of Elastic's copyrights. Elastic is further entitled to recover from floragunn the gains, profits, and advantages it has obtained as a result of its acts inducing, causing, encouraging, or materially contributing to Search Guard users' and third parties' infringement of Elastic's copyrights. The full extent of Elastic's damages and the gains, profits, and advantages floragunn has obtained by reason of its aforesaid acts of copyright infringement by Search Guard users and third parties cannot be determined at this time but will be proven at trial. Further, Elastic is entitled to recover costs and reasonable attorneys' fees from floragunn as a result of the acts inducing, causing, encouraging, or materially contributing to Search Guard users' and third parties' infringement of Elastic's copyrights alleged herein.

Response: floragunn denies the allegations in paragraph 95 of the amended complaint for the reasons set forth above in this Answer.

DEFENSES

Elastic's claims of copyright infringement concern a total of only about 100 out of more than 60,000 lines of source code that makes up the totality of floragunn's Search Guard security plugin. floragunn denies that any similarities are due to copying of any protectable expression original to Elastic, for at least the reasons stated below.

floragunn alleges and asserts the following defenses in response to the allegations contained in the amended complaint, undertaking the burden of proof only to the extent that they are deemed affirmative defenses by law. floragunn specifically reserves all rights to allege additional defenses and counterclaims that become known through its investigation into Elastic's allegations in the course of discovery.

First Defense – Independent Creation

1. The Search Guard security plugin was created independently.

2. floragunn's security plugin Search Guard traces its roots to October 2013, when Hendrik Saly, then an independent programmer, but now floragunn's Chief Technology Officer,

developed the first complete security plugin solution for the Elasticsearch search engine, appropriately called "Elasticsearch Security Plugin" ("ESP"). ESP was and is open code and has been publicly visible for all since 2013.

3. At the time, Elastic had not developed and was not offering a security plugin for its Elasticsearch search engine.

4. In January 2015, Mr. Saly began work on "Elastic Defender" ("Defender"), an advanced security plugin based on ESP.

5. In January 2015, Elastic finally released its own security plugin for Elasticsearch called "Shield." The source code for Shield was not open to the public, and Shield lacked many of the features that had been included in Defender, such as Kerberos, Field Level Security, Document Level Security, Index based output for audit events, Native Realm for storing users, and PKI authentication. In short, Shield was an objectively inferior security product when compared to ESP and its successor Defender.

6. That same month, January 2015, Shay Banon, the founder of Elastic, emailed Hendrik Saly, writing that he:

> *Just came across your Elasticsearch security plugin and we are looking for security and generally talented engineers with elastic knowledge to joining our company. Interested? Up for a quick chat?*

A period of interviewing ensued. As part of those discussions, in April 2015, Mr. Saly provided the source code for Defender to Mr. Banon and therefore to Elastic. Defender's source code had not been made publicly available at that point. Ultimately, Elastic did not offer Mr. Saly employment.

7. In May 2015, floragunn acquired an exclusive license from Mr. Saly for the Defender security plugin and set out to improve the code for the product before formally launching Defender rebranded as "Search Guard." floragunn made the source code for Defender / Search Guard available to the public on May 25, 2015.

8. In June 2015, two months after Mr. Saly provided the source code for Defender to Elastic, and a month after floragunn made the source code for Search Guard public, Elastic released Shield version 1.3.0, which for the first time contained a PKI authentication feature and an Index Output for audit event feature. These features had never been part of Shield before, but both had previously been included in Defender (the predecessor to floragunn's Search Guard). In October 2015, Elastic released Shield version 2.0, which for the first time, contained the features Field Level Security and Document Level Security features. These features had not been part of Shield in any previous release, but had previously been included in Defender.

9. There can be no doubt that Hendrik Saly and floragunn were the security plugin innovators, and Elastic the follower. Considering floragunn's expertise, and Elastic's late start in the security plugins, there was never any need for floragunn to copy Elastic's code since its security plugin development capabilities already exceeded Elastic's. Moreover, the idea that floragunn would copy code from Elastic as alleged in its amended complaint, and then publicly post that code for all to see, and therefore jeopardize its entire business and reputation defies logic. All of floragunn's code was independently created, and therefore does not infringe Elastic's code.

Second Defense – No Copyright Infringement

10. floragunn does not infringe, has not infringed (directly, contributorily, or by inducement), and is therefore not liable for infringement of any valid copyright or copyrights of Elastic, including, without limitations, any copyright rights in the works that are the subject of Plaintiff's Amended Complaint, including but not limited to the registered copyrights identified in paragraph 82 of the amended complaint.

11. Among other things, the Elasticsearch search engine and Kibana for which both the Shield (later X-Pack) and Search Guard plugins are created are based on open source code authored by others, such as Lucene, Netty, AngularJS and Node.js. Elasticsearch and Kibana

necessarily rely on a variety of code and syntax that was created by others. This means that substantial portions of the Elasticsearch and Kibana computer code are not original to Elastic.

12. Like Elasticsearch and Kibana, aspects of Shield and X-Pack are based on code not original to Elastic, including but not limited to open source libraries or code such as Lucene, Netty, AngularJS, Lodash, and Node.js, and since such aspects are not Elastic's original expression, they are not entitled to copyright protection.

Third Defense -- Elements Not Protected by Copyright

13. Elasticsearch is designed specifically to allow the creation of plugins like Shield and Search Guard to extend its functionality. Elastic has long documented the specific means that developers should follow when creating Elasticsearch plugins. The requirements for a plugin to operate necessarily constrain the choices made by plugin developers. This is analogous to third-party upgrade kits for physical appliances; for example, a turbocharger for a car necessarily will need to physically mate with the engine of the car, and thus the design choices for a turbocharger for a Corvette will necessarily be constrained by the physical design of a Corvette. Search Guard likewise must behave in certain ways in order for it to operate with Elasticsearch.

14. Elastic's own Shield product likewise is a plugin for Elasticsearch, and thus the design choices for Shield were also constrained by the need to operate with Elasticsearch. Just as two turbochargers designed to work with a Corvette will share certain similarities because both must attach to the same engine, Search Guard and Shield likewise must share some similarities because both are plugins for the same Elasticsearch product.

15. Furthermore, Search Guard and Shield (X-Pack) both provide similar functionality for certain features (though the Search Guard product offers more functionality than Elastic's security plugin). That similarity in functionality acts as a further constraint on the design choices that floragunn and Elastic respectively made in designing their products.

Returning to the turbocharger analogy, any turbocharger necessarily will include a turbine—that is because turbochargers use turbines, not because one turbocharger is a copy of another.

16. In addition, there are many "tools of the trade" that are known to many in the developer community, and that draw on tropes common to computer programming. These tropes can result in superficial similarities between independently developed code, especially when the code is reviewed by a lay observer who is not familiar with the common programming conventions that are known to software engineers.

17. Elastic's copyright claims are barred to the extent that Elastic claims rights to elements of Elastic's software or other works that are functional, are not original, or are otherwise not protectable by copyright or are otherwise not protected by the registered copyrights identified in paragraph 82 of the Amended Complaint.

Fourth Defense – Fair Use

18. Elastic's claims for copyright infringement are barred in whole or part by the doctrine of fair use pursuant to 17 U.S.C. § 107 in view of the nature of works asserted by Elastic and covered by the copyrights identified in paragraph 82 of the Amended Complaint, the amount (if any) and substantiality of the portions of such works used by floragunn (if any), in relations to the works as a whole, the purpose and character of any use thereof by floragunn, and the effect, if any, of such use on the potential market for the works.

Fifth Defense – *De Minimis* Copying

19. Elastic's claims for copyright infringement are barred by the doctrine of de minimis copying, as any alleged copying of protectable portions of the work that are the subject of the claimed copyrights was *de minimis*.

Sixth Defense – No Intent to Induce Copyright Infringement

20. floragunn has not engaged in purposeful, culpable expression or conduct designed or intended to result in others infringing Elastic's alleged copyrights and thus is not liable under Elastic's inducement claims.

Seventh Defense -- No Injunctive Relief

21. Elastic has not suffered any irreparable injury, and has an adequate remedy at law, and injunctive relief is unwarranted because it would be contrary to the public interest.

Eighth Defense – Statute of Limitations

22. Elastic's claims for damages are barred in part by the applicable statute of limitations.

COUNTERCLAIMS

As and for its counterclaims against plaintiff Elastic, floragunn alleges as follows:

Parties

1. floragunn GmbH ("floragunn") is a corporation organized and existing under the laws of Germany with its principal place of business in Germany. Upon information and belief, Elasticsearch, Inc. is incorporated in Delaware with a principal place of business in Mountain View, California. Plaintiff elasticsearch B.V. is incorporated in the Netherlands, with a principal place of business in the Netherlands. (Elasticsearch, Inc. and elasticsearch B.V. are hereinafter "Elastic".)

Jurisdiction and Venue

2. Subject to floragunn's defenses and denials, floragunn alleges that this Court has jurisdiction over the subject matter of these Counterclaims pursuant to 28 U.S.C. §§ 1331, 1338(a), 1367, 2201, and 2202, and venue for these Counterclaims is proper in this district.

3. This Court has personal jurisdiction over Elastic.

Factual Background

4. Elasticsearch is a search and analytics engine owned and operated by Elastic that allows users to build upon it and search out their own data.

5. Elasticsearch is designed specifically to allow the creation of plugins to extend its functionality.

6. Elastic has long documented the specific means that developers should follow when creating Elasticsearch plugins.

7. Kibana is Elastic's user interface designed to manage and configure Elasticsearch and other Elastic products and to produce data visualizations including diagrams and dashboards.

8. Kibana is designed specifically to allow the creation of plugins to extend its functionality.

9. Elastic has long referred interested plugin authors to publicly availably posts for the specific means that developers should follow when creating Kibana plugins.

10. Elasticsearch and Kibana are themselves based on code not original to Elastic, including but not limited to Lucene, Netty, AngularJS, Lodash, and Node.js.

11. floragunn's Search Guard is a security plugin for Elasticsearch and Kibana.

12. Elastic does not claim that floragunn's Search Guard infringes the Elasticsearch search engine.

13. Elastic does not claim that floragunn's Search Guard infringes Kibana.

14. Elastic claims that Search Guard infringes Shield (later the security plugin for Elastic's X-Pack).

15. Elastic's Shield is Elastic's security plugin for Elasticsearch and Kibana.

16. Like Elasticsearch and Kibana, Shield is based on code not original to Elastic, including but not limited to open source libraries or code such as Lucene, Netty, AngularJS, Lodash, and Node.js.

17. Shield and Search Guard are competitive plugin products for Elasticsearch and Kibana.

18. Many aspects of Shield and Search Guard source code are constrained by choices made by the programmers who wrote code that is not original to Elastic, such as Lucene, Netty, AngularJS, Lodash, and Node.js. Many aspects of Shield and Search Guard source code are constrained by the need for each of Shield and Search Guard to function as a plugin to Elasticssearch and/or Kibana.

19. Design choices made by Elastic for Shield are constrained by the need to operate with Elasticsearch and with Kibana.

20. Design choices made by floragunn for Search Guard are constrained by the need to operate with Elasticsearch and with Kibana.

21. The source code for Search Guard and Shield share similarities because both are plugins for the same Elastic products (Elasticsearch and Kibana).

22. Aspects of Elastic's source code in Shield that Elastic claims was infringed are not original to Elastic but are derived from open source libraries or licenses.

23. Aspects of floragunn's code alleged by Elastic to infringe Shield or X-Pack concern standard, common, or stock programming practices.

24. There are many "tools of the trade" that are known to many in the developer community, and that draw on tropes common to computer programming. These tropes can result in superficial similarities between independently developed code, especially when the code is reviewed by a lay observer who is not familiar with the common programming conventions that are known to software engineers.

25. Certain similarities in the Shield and Search Guard source code identified in the complaint are superficial similarities between independently created code.

26. floragunn's Search Guard does not infringe Elastic's source code for Shield because Search Guard was independently created, because portions of the alleged infringed code are not Elastic's original expression, because portions of the alleged infringing code are designed specifically to extend Elasticsearch's functionality and the choices made by floragunn in the creation of Search Guard were constrained by requirements of operating as a plugin to Elasticsearch and Kibana; because portions of the alleged infringing code are common tropes well known and used in the software developer community; or because the alleged infringed code is not otherwise protected by copyright.

COUNT ONE

Declaratory Judgment of Non-Infringement of Asserted Copyrights

27. floragunn restates and incorporates by reference its allegations in paragraphs 1 to 26 of its Counterclaims.

28. An actual case or controversy exists between floragunn and Elastic as to whether the Asserted Copyrights are infringed by floragunn.

29. A judicial declaration is necessary and appropriate so that floragunn may ascertain its rights regarding the Asserted Copyrights.

30. floragunn has not infringed and does not infringe, directly or indirectly, the Asserted Copyrights, nor has floragunn contributed to any infringement by any third parties of Elastic's copyrights.

COUNT TWO

17 U.S.C. § 512(f) Misrepresentation

31. floragunn restates and incorporates by reference its allegations in paragraphs 1 to 30 of its Counterclaims.

32. floragunn's security plugin Search Guard traces its roots to October 2013, when Hendrik Saly, then an independent programmer, but now floragunn's Chief Technology Officer,

developed the first complete security plugin solution for the Elasticsearch search engine, appropriately called "Elasticsearch Security Plug-in" ("ESP").

33. ESP was and is open source and its source code has been publicly visible for all since 2013.

34. In 2013, Elastic had not developed and was not offering a security plugin for its Elasticsearch search engine.

35. In January 2015, Mr. Saly began work on "Elastic Defender" ("Defender"), an advanced security plugin based on ESP.

36. In January 2015, Elastic for the first time released its own security plugin for Elasticsearch called "Shield."

37. The source code for Shield was not open to the public when it was released.

38. Shield lacked many of the features that had been included in Defender, such as Kerberos, Field Level Security, Document Level Security, Index based output for audit events, Native Realm for storing users, and PKI authentication.

39. In January 2015 Shield was an objectively inferior security product when compared to ESP and its successor Defender.

40. That same month, January 2015, Shay Banon, the founder of Elastic, emailed Hendrik Saly, writing that he:

> *Just came across your Elasticsearch security plugin and we are looking for security and generally talented engineers with elastic knowledge to joining our company. Interested? Up for a quick chat?*

41. A period of interviewing ensued.

42. As part of those discussions, in April, 2015, Mr. Saly provided the source code for Defender to Mr. Banon and therefore to Elastic.

43. Defender's source code had not been made publicly available at that point.

44. Elastic ultimately did not offer Mr. Saly employment.

45. Upon information and belief, one or more of Elastic's agents or employees reviewed the Defender Source Code provided to Mr. Banon by Mr. Saly.

46. Prior to commencing this action in September 2019, at no time did Elasticsearch notify Mr. Saly that the code he provided to Elastic infringed any code of Elastic.

47. Prior to the date on which it commenced this action in September 2019, at no time did Elastic send a takedown notice to any host of Mr. Saly's code pursuant to the Digital Millennium Copyright Act ("DMCA") or otherwise.

48. In May 2015, floragunn acquired an exclusive license from Mr. Saly for the Defender security plugin and set out to improve the code for the product before formally launching Defender rebranded as "Search Guard."

49. floragunn made the source code for Defender / Search Guard available to the public on or about May 25, 2015.

50. Upon information and belief, Elastic was aware as early as 2015 that floragunn acquired the exclusive rights to Defender.

51. In June 2015, two months after Mr. Saly provided the source code for Defender to Elastic, and a month after floragunn made the source code for Search Guard public, Elastic released Shield version 1.3.0, which for the first time contained a PKI authentication feature and an Index Output for audit event feature.

52. The PKI authentication feature and Index Output feature had never been part of Shield prior to version 1.3.0, but both had previously been included in Defender (the predecessor to floragunn's Search Guard).

53. In October 2015, Elastic released Shield version 2.0, which for the first time, contained the features Field Level Security and Document Level Security.

54. These features (Field Level Security and Document Level Security) had not been part of Shield in any previous release, but had previously been included in Defender (the predecessor to Search Guard).

55. Upon information and belief, by October 2015, Elastic had access to and in fact reviewed floragunn's Search Guard source code which was available and open to the public.

56. November 2016, Elastic's founder Shay Banon personally sent an email to floragunn's founders stating to "express [his] deep concerns" and claiming that "Search Guard appears to be based upon Elastic's 'Shield.'"

57. Mr. Banon did not provide floragunn with any examples of allegedly infringing code that he claimed was "based on Elastic's 'Shield'" but demanded that floragunn "stop distributing Search Guard and providing any support for it to those end users that already have it" and threatened litigation if floragunn did not do what he asked.

58. Thus, by November 2016, Elastic had reviewed floragunn's Search Guard source code to ascertain whether it contained infringing code.

59. Notwithstanding its allegations of infringement, at no time prior to the date on which it commenced this action in September 2019 did Elastic send a DMCA takedown notice to the host of floragunn's code, nor did Elastic file any legal action against floragunn before then or seek any injunctive relief related to any alleged copyright infringement.

60. In January 2017, Elastic's German counsel, Osborne Clarke, sent a letter to floragunn making unspecified claims to floragunn stating that "you use in your software 'Search Guard' part of Elasticsearch-Software," further demanded that that floragunn "assure us that your company is not using our client's software codes, or if you are using them why you feel that your company has the right to do so."

61. In its January 2017 letter, Osborne Clarke gave no specifics about what source code it claims was infringing or was being infringed.

62. In February 2017, floragunn's German counsel responded to the Osborne Clarke letter, writing that that they did not elaborate on which program parts were involved and that they did not take into account the fact that essential parts of the software at issues are licensed under open source.

63. Elastic's German Counsel did not respond to floragunn's invitation to provide specific information concerning of what exactly Elastic was claiming floragunn was infringing.

64. Upon information and belief, by the time the instant action was commenced on September 4, 2019, Elastic had been reviewing floragunn's source code for Search Guard available on GitHub and Sonatype for more than four years.

65. All of the Search Guard source code, including modification, additions, fixes and commits, was available for public view continuously since May, 2015 in repositories hosted on GitHub and Sonatype.

66. On or about September 10, 2019, Elastic sent DMCA takedown notices to GitHub and to Sonatype stating that it had a good faith belief that use of the material in the manner complained of is not authorized by the copyright owner, its agent, or the law.

67. On information and belief, having had four years to review floragunn's source for Search Guard on GitHub and Sonatype, and having reviewed the Search Guard publicly available code in fine detail for the

purpose of commencing the instant action, Elastic was aware that a substantial portion of the content in the repositories hosted on GitHub and Sonatype did not contain materials that infringed Elastic's code.

68. Elastic had access to and on information and belief in fact, inspected all of floragunn's open code posted on Sonatype and GitHub prior to sending out a DMCA takedown notices to Sonatype and GitHub.

69. As part of its DMCA notice to Sonatype, Elastic included in its 73-page long list of claimed "Infringing Content to Be Removed" files that did not contain any content that Elastic claims to be infringing.

70. Annexed hereto as Exhibit A is a list of files that were taken down by Sonatype because they were specifically identified in Elastic's DMCA takedown notice to Sonatype as "infringing content". None of the files listed in Exhibit A contain content alleged to infringe Elastic's copyrights.

71. Similarly, as part of its DMCA take down notice to GitHub, Elastic specifically identified the repository located at https://github.com/floragunncom/search-guard-ssl as containing infringing content, which resulted in DMCA taking down all content in that repository.

72. Annexed hereto as Exhibit B is a list of files that were taken down by GitHub because they were included in the repository identified in DMCA takedown notice to GitHub as containing infringing material. None of the files listed in Exhibit B contain content alleged to infringe Elastic's copyrights.

73. On information and belief, Elastic knew that at least some of the content included identified or covered by its DMCA takedown notices to GitHub and Sonatype did not infringe any Elastic copyright.

74. On information and belief, Elastic at a minimum knew that there was a high probability that at least some of the content and material listed in Exhibits A and B did not infringe any Elastic copyright, and took deliberate actions to avoid learning that at least some of such content did not infringe any Elastic copyright.

75. On information and belief, at the time Elastic sent the takedown notices, Elastic knew that at least some of the material covered by its DMCA notices to Sonatype and GitHub did not contain infringing material, and therefore knowingly misrepresented in the DMCA

notices it sent that the material it sought to take down did not violate Elastic's copyrights. On information and belief, Elastic's misrepresentations were material to the decision by GitHub and/or Sonatype to remove or disable access to at least some of the content and material listed in Exhibits A and B.

76. Congress imposed liability on a copyright owner who "knowingly materially misrepresents" that the material that it requests to be taken down is infringing.

77. Section 512(f) of the DMCA provides in pertinent part that:

> Any person who knowingly materially misrepresents under this section— (1) that material or activity is infringing, or (2) that material or activity was removed or disabled by mistake or misidentification, shall be liable for any damages, including costs and attorneys' fees, incurred by the alleged infringer, by any copyright owner or copyright owner's authorized licensee, or by a service provider, who is injured by such misrepresentation, as the result of the service provider relying upon such misrepresentation in removing or disabling access to the material or activity claimed to be infringing, or in replacing the removed material or ceasing to disable access to it.

78. On information and belief, Elastic knew that at least some of the content and materials listed in Exhibits A and B did not infringe any of Elastic's copyrights on the dates that they sent GitHub and Sonatype takedown notices under the DMCA.

79. In addition, upon information and belief, the fact that Elastic had full access to floragunn's source code on GitHub and Sonatype for more than four years, but delayed sending DMCA takedown notices concerning any such material despite having sent non-specific correspondence to floragunn alleging infringement, evidences the fact that Elastic did not believe that any of floragunn's material hosted by GitHub and Sonatype was infringing material.

80. Accordingly, Elastic violated 17 U.S.C. §512(f) by knowingly and materially misrepresenting that the content and material listed in Exhibits A and B infringed Elastic's rights.

81. As a direct and proximate result of Elastic's actions, floragunn has been injured, including but not limited to the interruption of floragunn's business and the expenses, as well as costs and attorneys' fees.

PRAYER FOR RELIEF

WHEREFORE, having fully responded to Elastic's Amended Complaint and asserted its Counterclaims against Elastic, floragunn prays for judgment as follows:

a. A judgment dismissing Elastic's Amended Complaint against floragunn with prejudice;

b. A judgment in favor of floragunn on all its Counterclaims;

c. A declaration that floragunn has not infringed, contributed to the infringement of, or induced others to infringe, either directly or indirectly, any of Elastic's claimed copyrights.

d. A declaration that Elastic's claims are barred because Elastic's claimed rights to elements of Elastic's software or other works not protected by copyright because they are not original, or are otherwise not protectable by copyright or are not otherwise protected by the registered copyright identified in paragraph 82 of the Amended Complaint.

e. An award to floragunn of its reasonable costs and expenses of litigation, including attorneys' fees; and

f. Such other and further relief as this Court may deem just and proper.

DATED: December 24, 2019

WUERSCH & GERING LLP

By: /s/ V. David Rivkin /s/
V. DAVID RIVKIN (admitted *pro hac vice*)
david.rivkin@wg-law.com
JOHN A. SMITTEN (admitted *pro hac vice*)
john.smitten@wg-law.com
100 Wall St., 10th Fl.
New York, NY 10005
Telephone: 212 509-5050
Facsimile: 212 509-9559

KWUN BHANSALI LAZARUS LLP
MICHAEL S. KWUN (SBN 198945)
mkwun@kblfirm.com
555 Montgomery St., Suite 750
San Francisco, CA 94111
Telephone: 415 630-2350
Facsimile: 415 367-1539

Attorneys for Defendant
FLORAGUNN GmbH

DEMAND FOR JURY TRIAL

In accordance with Rule 38 of the Federal Rules of Civil Procedure and Civil L.R. 3-6(a), floragunn GmbH respectfully demands a jury trial of all issues triable to a jury in this action.

DATED: December 24, 2019

WUERSCH & GERING LLP

By: /s/ V. David Rivkin /s/
V. DAVID RIVKIN (admitted pro hac vice)
david.rivkin@wg-law.com
JOHN A. SMITTEN (admitted pro hac vice)
john.smitten@wg-law.com
100 Wall St., 10th Fl.
New York, NY 10005
Telephone: 212 509-5050
Facsimile: 212 509-9559

KWUN BHANSALI LAZARUS LLP
MICHAEL S. KWUN (SBN 198945)
mkwun@kblfirm.com
555 Montgomery St., Suite 750
San Francisco, CA 94111
Telephone: 415 630-2350
Facsimile: 415 367-1539

Attorneys for Defendant
FLORAGUNN GmbH

CERTIFICATE OF SERVICE

I am an attorney at Wuersch & Gering LLP, counsel for Defendant, in the above-captioned proceeding. I hereby certify that on December 24, 2019, I caused the foregoing **Answer to First Amended Complaint with Counterclaims** to be served electronically via CM/ECF upon Plaintiffs Elasticsearch, Inc. and elasticsearch, B.V.

/s/ John A. Smitten /s/
John A. Smitten