CASSANDRA ve MongoDB NoSQL VERİTABANLARININ KARŞILAŞTIRILMALI GÜVENLİK ANALİZİ

Murat SARAN ve Nuran SARAN
Çankaya Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Ankara

ÖZET

Bu çalışmada, MongoDB 3.6.3 ve Cassandra 3.11.1 NoSQL veri tabanlarının güvenliğinin çok düğümlü yapılandırmada ve iki adımda karşılaştırılmalı analiz sonuçları sunulmaktadır. İlk adımda, her iki veritabanının güvenlik özelliklerinin literatürden seçilen on farklı ölçüte göre karşılaştırmalı bir analizi gerçekleştirilmiştir. İkinci adımda ise, Yahoo Cloud Serving Benchmark aracını kullanarak her iki NoSQL veri tabanının veri şifreleme ve şifre çözümleme performansları karşılaştırılmıştır. Bu çalışma ile karar vericilere ve araştırmacılara NoSQL veri tabanlarının güvenlik özellikleriyle ilgili analiz sürecinde yol göstermek ve NoSQL veri tabanlarıyla ilgili en önemli güvenlik özelliklerini ortaya çıkarmak amaçlanmıştır. Güvenlik karşılaştırma sonuçları, her iki NoSQL veri tabanının da kayda değer güvenlik özelliklerine sahip olduğunu göstermektedir. Bununla birlikte, Cassandra'nın daha fazla güvenlik ölçütünü desteklediği ortaya çıkmıştır. Ayrıca, çalışma zamanı ve performans ile ilgili yapılan karşılaştırmada MongoDB kurumsal sürümünün şifreleme/şifre çözümleme performansının Cassandra kurumsal sürümünden ortalama %53 daha hızlı olduğu ve dakikada işleyebildiği veri miktarının ortalama %45 daha fazla olduğu bulunmuştur. Bu sonuç şifrelemenin bir gereklilik olduğu ortamlarda MongoDB'nin kullanılmasının daha uygun olduğunu göstermektedir.

Anahtar Kelimeler— NoSQL veri tabanı, güvenlik, karşılaştırma, Cassandra, MongoDB

A Comparative Security Analysis of Cassandra and MongoDB NoSQL Databases

ABSTRACT

In this study, we analyze the security of two NoSQL databases, MongoDB 3.6.3 and Cassandra 3.11.1 in a multi-node configuration in two steps. The first step is a comparative study of both databases' security features according to ten selected criteria from the literature. The second step is analyzing data encryption overhead using the Yahoo Cloud Serving Benchmark tool. This study will help decision-makers and researchers to realize the most crucial security features concerning NoSQL databases as well as to be able to analyze the NoSQL databases regarding the security features. Our security comparison results show that both databases have noteworthy security features. However, Cassandra takes the lead as it supports more security criteria. Besides, the encryption/decryption performance of the MongoDB business version is 53% faster than the Cassandra business version, and the average amount of data that the MongoDB business version can process per minute is 45% higher than the Cassandra business version. This result shows that it is more appropriate to use MongoDB in environments where encryption is required.

Keywords— NoSQL database, security, comparison, Cassandra, MongoDB

I. GİRİŞ (INTRODUCTION)

Özellikle sosyal platformlar ve sensörler aracılığıyla oluşan ve yüksek bir hızla artan büyük verilerle uğraşmak, başta NoSQL veri tabanları olmak üzere çeşitli çekirdek teknolojilerin ve veri tabanı platformlarının geliştirilmesine yol açmıştır. Her ne kadar NoSQL veri tabanları büyük verilerin ortaya çıkmasından önce gelişmeye başlasa da NoSQL veri tabanlarının benimsenmesi büyük verilerin saklanması ve bu verilere İnternet ortamında

erişilmesi ihtiyacı ortaya çıkıncaya kadar gerçekleşmemiştir. Her biri kendine özgü mimari tasarıma, sıkıştırma yöntemine, güvenlik özelliklerine, kümeleme ve ölçeklenebilirlik desteğine ve sorgu diline sahip birçok farklı NoSQL veritabanı ortaya çıkmıştır ve seçeneklerin çok olması ihtiyaca uygun NoSQL veri tabanına karar verilmesini zorlaştırmaktadır [1][2]. Piyasada 245'ten fazla farklı NoSQL veritabanı sistemi mevcuttur [3]. Bu nedenle, uygun veri tabanını seçmek zor ve karmaşık bir görevdir. Çünkü bu veri tabanları üzerinde sorgu işleme hızı, veri güvenliği analizi ve ölçeklenebilir bir ortamda kullanmaya hazır olma gibi çeşitli faktörlerle ilgili incelemeler yapılmalıdır. Literatürde birçok çalışmada performans, ölçeklenebilirlik ve diğer özellikler ile ilgili birçok NoSQL veri tabanı sisteminin karşılaştırılmasına rağmen [4][5][6], NoSQL veri tabanlarının güvenlik özelliklerini ve şifreleme performansını birlikte inceleyen çalışmaya rastlanmamıştır. Bu nedenle, bu çalışma yaygın olarak kullanılan iki NoSQL veri tabanının (MongoDB ve Cassandra) kapsamlı bir güvenlik ve performans analizini kapsamaktadır [7]. Bu analiz sonucunda karar vericilere ve araştırmacılara NoSQL veri tabanlarının güvenlik özellikleriyle ilgili inceleme çalışmalarında yol göstermek ve NoSQL veri tabanlarıyla ilgili en önemli güvenlik özelliklerini ortaya çıkarmak amaçlanmıştır.

Bu çalışmada, MongoDB 3.6.3 ve Cassandra 3.11.1'in güvenliğinin çok düğümlü bir yapılandırmada ve iki adımda analiz sonuçları sunulmaktadır. MongoDB ve Cassandra'nın tek düğümlü testi, her iki sistemin de performans ve işlem gecikmeleri açısından nerede durduğunu ortaya koymakla birlikte, böyle bir test sonucu bir küme yapılandırmasındaki aynı veri tabanlarının performansı için genelleştirilemez. Bununla birlikte, NoSQL veri tabanları gerçek hayatta genellikle birden fazla sunucu üzerinde dağıtık bir yapıda kullanılmaktadır. Bu nedenle, bu çalışmada her iki veritabanının performans testinin bir küme yapılandırmasında analiz etmenin gerekli olduğu düşüncesiyle çok düğümlü yapılandırma testi gerçekleştirilmiştir. İlk adımda, her iki NoSQL veri tabanının güvenlik özelliklerinin literatürden [8][9][10][11][12] seçilen on farklı ölçüte göre karşılaştırmalı bir analizi gerçekleştirilmiştir. İkinci adımda ise, Yahoo Cloud Serving Benchmark (sürüm YCSB-0.12.0) aracını kullanarak her iki NoSQL veri tabanının veri şifreleme ve şifre çözümleme performansları karşılaştırılmıştır [13].

Bir uygulamada hangi NoSQL veri tabanının kullanılacağını belirleyen ilk faktör performans ise, bilgi güvenliği ikincil ya da eş önemli faktör olarak kabul edilebilir. Ancak, veri tabanlarındaki güvenliği belirlemek veya ölçmek için özel araçlar olmadığından, bilgi güvenliği açısından hangi veri tabanının nispeten daha iyi olduğunu belirlemeye yarayan çeşitli güvenlik faktörlerini bilmek ve farkında olmak önemlidir. Buna ek olarak, veri tabanlarının şifreleme performansı veri tabanı seçim sürecinde büyük bir öneme sahiptir. Çünkü şifreleme genellikle bir kaynak tüketim sürecidir. Bu süreçte veri tabanı altyapısı gelen tüm veri akışlarını şifrelemek ve şifreli sorguları çözmek için sistem kaynaklarını yoğun bir şekilde kullanır. Bu nedenle, her iki veri tabanının şifreleme motorunun karşılaştırmalı performans testini gerçekleştirmek literatüre önemli katkı sağlayabilir.

Büyük verilerin ortaya çıkmasıyla NoSQL veri tabanlarının geniş bir şekilde benimsenmesi, NoSQL veri tabanlarının yapılandırılmamış verileri, ölçeklenebilirliği ve performansı etkili bir şekilde yönetme yeteneğinden kaynaklanmaktadır. Ancak, NoSQL veri tabanı geliştiricilerinin çoğu, başlangıçta güvenlik pahasına performansa odaklanmıştır [8][14]. Bu nedenle, NoSQL güvenliği bazı araştırma ve incelemelere konu olmuştur. Örneğin, Cassandra ve MongoDB'nin veri şifreleme, kimlik doğrulama, yetkilendirme, sorgu dili ve denetleme gibi güvenlik özelliklerini incelemiştir [8]. Araştırmacılar bu çalışmada her iki veri tabanının da hizmet reddi saldırıları ve çok basit yetkilendirme mekanizmalarına karşı savunmasız olduğunu ve zayıf kimlik doğrulama ile zayıf veri şifreleme özelliklerine sahip olduğu sonucuna varmıştır. Buna ek olarak, 2014 yılında altı NoSQL veri tabanının beş güvenlik özelliği kullanılarak analiz edildiği bir başka çalışma yayınlanmıştır [9]. Çalışma sonuçları Tablo 1'de sunulmuştur. Araştırmacılar, NoSQL veri tabanlarının güvenlik özelliklerinin geliştirilmeye ihtiyacı olduğu sonucuna varmışlardır.

2015'te üç NoSQL veri tabanı ve iki ilişkisel veri tabanının karşılaştırıldığı çalışmada, gizlilik, bütünlük ve kullanılabilirlik açısından güvenlik analizi yapılmıştır [15]. Araştırma, NoSQL veri

tabanlarının güvenlik konusunda yüksek hızla geliştiği sonucuna varmıştır. Ayrıca, açık kaynaklı NoSQL veri tabanlarının hızla çoğalması ve yaygınlaşmasıyla, güvenliğin ilişkisel veri tabanlarıyla karşılaştırılabilir bir seviyeye ulaştığını öne sürmüşlerdir. Son olarak, 2017'de yayınlanan bir çalışmada araştırmacılar, NoSQL veri tabanlarındaki çeşitli güvenlik kaygılarını daha da iyileştirmek için veri maskeleme tekniğini önermişlerdir [16].

Tablo 1. Çeşitli NoSQL veri tabanlarındaki güvenlik özelliklerinin karşılaştırmalı analizi [9]
(Comparative analysis of security features in various NoSQL databases)

Ölçüt (Criterion)	Veri tabanı (Database)					
	MongoDB	Redis	CouchDB	Cassandra	HBase	Couchbase
Kimlik Doğrulama (Authentication)	Orta (Medium)	Zayıf (Low)	Orta (Medium)	Zayıf (Low)	Orta (Medium)	Orta (Medium)
Erişim kontrolü (Access Control)	Güçlü (High)	Zayıf (Low)	Zayıf (Low)	Zayıf (Low)	Orta (Medium)	Zayıf (Low)
Güvenli Yapılandırma (Secure Configuration)	Orta (Medium)	Zayıf (Low)	Zayıf (Low)	Zayıf (Low)	Zayıf (Low)	Zayıf (Low)
Veri şifreleme (Data Encryption)	Orta (Medium)	Zayıf (Low)	Orta (Medium)	Orta (Medium)	Zayıf (Low)	Zayıf (Low)
Denetleme (Auditing)	Zayıf (Low)	Zayıf (Low)	Orta (Medium)	Zayıf (Low)	Orta (Medium)	Orta (Medium)

Veri maskeleme, veri tabanları içinde hassas veya gizli verilere erişimi engelleyen bir güvenlik yöntemidir. Bu, gerçek verileri gerçek olmayan ancak uygun verilerle değiştirerek gerçekleştirilir. Bu yöntemin amacı, bir organizasyonun içinde ve dışında hassas bilgilerin sızması ve ele geçirilmesi riskini azaltmaktır. Araştırmacılar, bu yöntemin NoSQL veri tabanı güvenliğinin artırılmasına anlamlı bir katkı sağladığını öne sürmüşlerdir.

Literatürde NoSQL veri tabanlarının başlıca avantajlarının yüksek performans, ölçeklendirme kolaylığı, esneklik ve ağ trafiğini azaltma ve dolayısıyla ağ maliyetinde azalma olduğu belirtilmektedir [17]. Ancak, gittikçe daha hassas olan veriler NoSQL veri tabanlarında depolandığından, davetsiz misafirlerden gelen erişim çabaları veri tabanının güvenliğini giderek artan bir endişe haline getirmektedir. Bu nedenle, NoSQL veritabanı ortamlarının güvenlik sorunları, bütünlüğün korunması ve erişim kontrolü dahil olmak üzere birden fazla alanda incelenmelidir. Bu nedenle, bu çalışmada kullanılan güvenlik analiz ölçütleri NoSQL veri tabanları için bütüncül bir analiz olanağı sunma hedefiyle belirlenmeye çalışılmıştır. Makalenin geri kalan bölümleri şu şekilde düzenlenmiştir: 2. Bölümde kullanılan deneysel metot sunulmuştur. 3. Bölümde analiz sonuçları verilmiştir. Son olarak, 4. Bölümde ise elde edilen sonuçlar literatürde yapılan önceki çalışmaların sonuçları ile karşılaştırılmış ve öneriler üzerinde durulmuştur.

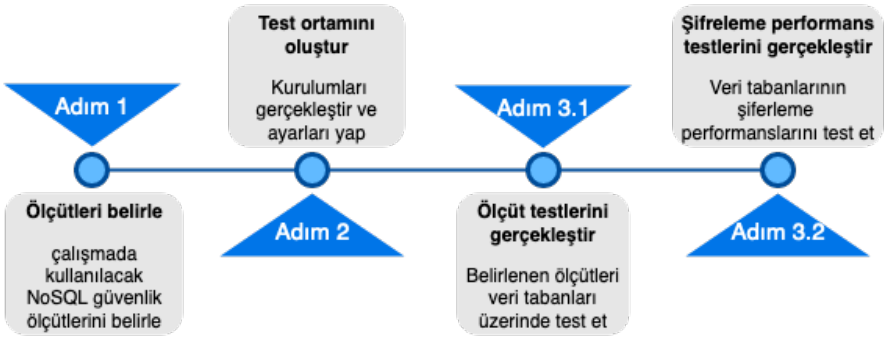
II. DENEYSEL METOT (EXPERIMENTAL METHOD)

Bu bölümde test ortamı, çalışmada kullanılan NoSQL veri tabanı güvenlik ölçütleri ve testlerin nasıl gerçekleştirildiği anlatılmaktadır. Bu çalışma sonuçları bu bölümde anlatılan test ortamı ve yazılımlar ile sınırlıdır. Çalışmanın sonuçlarını kullanırken bu husus göz önünde bulundurulmalıdır. Bu çalışmada 3 adımdan oluşan bir metot kullanılmıştır. Şekil 1'de çalışmada takip edilen metot gösterilmektedir. İlk adımda önce literatürde NoSQL veri tabanlarının güvenliğini inceleyen çalışmalar derlenmiş ve bu çalışmalarda kullanılan ölçütler belirlenmiştir. Daha sonra bu çalışmalarda yer alan tüm ölçütler birleştirilmiş ve bu çalışmada kullanılacak ölçütler seçilmiştir. İkinci adımda test ortamı oluşturulmuştur. Son adım iki basamakta gerçekleştirilmiştir. Birinci basamakta bu çalışmada kullanılan NoSQL veri tabanları üzerinde belirlenen her bir ölçüt test edilmiştir. İkinci basamakta ise şifreleme performans testleri gerçekleştirilmiştir. Aşağıda bu adımlar açıklanmıştır.

2.1. NoSQL Veri Tabanı Güvenlik Ölçütleri

NoSQL veri tabanı sistemleri başlangıçta güvenlik yerine performans odaklı tasarlanmışlardır. Ancak, kullanıcılar büyük

verilerin güvenlik zafiyetleri hakkında geri bildirimlerde bulunmaya başladığında, geliştiriciler güvenlik özelliklerini göz önünde bulundurmaya başlamışlardır. İlişkisel veri tabanı yönetim sistemlerinden farklı olarak, güvenlik NoSQL veri tabanlarının performansıyla direk bağlantılıdır. Sistemlere güvenlik özellikleri eklendikçe performans düşmektedir. Karar vericiler bu hususu özellikle dikkate almalıdır. Bu bölümde NoSQL veri tabanlarının güvenlik analizinde kullanılabilecek ve bu çalışmada kullanılan 10 farklı güvenlik ölçütü açıklanmıştır. Bu ölçütler literatürde incelenen NoSQL veri tabanlarının en önemli güvenlik özelliklerini kapsamaktadır.



Şekil 1. Çalışmada kullanılan metot (Method used in the study)

2.1.1. Kimlik doğrulama (Authentication)

Başlangıçta, NoSQL veri tabanları kimlik doğrulama için uygun desteği içermemekte idi [18]. Ancak, günümüzde NoSQL veri tabanlarının çoğu kimlik doğrulama özelliğini sunmaktadır. Bu işlev, kullanıcı verilerine ancak doğru kullanıcı adı ve parola bilindiğinde erişilebilmesinden sorumludur.
Bu bölümün ana faaliyetleri aşağıdaki gibidir:

- Kullanıcıya farklı kullanıcılarla aynı giriş bilgilerini oluşturmaktan kaçınmak için seçtiği kullanıcı adının kullanılabilirliği hakkında bilgi sağlamak.
- Veri tabanının kullanıcılarına merkezi erişim sağlama. Ayrıca, yetkisiz kullanıcının sisteme giriş yapmasını önlemek.
- Uygun bir güvenli parola oluşturmak için gerekenler hakkında kısıtlamalar sunmak. Parola karmaşıklık düzeyi için parolanın minimum içermesi gereken karakterler gibi kısıtlamalar sunulabilir.

2.1.2. Yetkilendirme (Authorization)

Yetkilendirme, kullanıcılara veri tabanının sadece izin verilen kısmına erişilebilirlik sağlama işlemidir, bu da kullanıcılara kısıtlamalar ve roller oluşturarak kullanıcıların veri tabanındaki nesnelere sınırlı erişimini sağlama anlamına gelir. Diğer kullanıcılar için gereksiz veya özel verileri göstermeden, tüm kullanıcılar için yalnızca erişilebilirlik hakkına sahip olduğu verilere ulaşmalarına olanak sağlar.

2.1.3. Denetleme (Auditing)

Denetleme, sistemdeki kullanıcı aktivitesini kaydeden ve gerektiğinde onları izleyen sistem mekanizması olarak tanınabilir. Ayrıca, kullanıcının parolalarındaki olası anormal durumları tespit etmede güvenliğe yardımcı olur. Denetim, aynı zamanda veri tabanına yetkisiz veri girme girişimini ve bunun nasıl yapıldığını tespit etme yeteneğidir. Denetim güvencesi, tüm isteklerin veri tabanı günlüklerine kaydedilmesi ile yapılabilir. Denetleme, sadece verideki değişiklikleri değil, veri tabanlarının konfigürasyon dosyalarındaki değişiklikleri de kapsamalıdır.

2.1.4. Aktarım şifreleme (Transport encryption)

Veriler kullanıcıdan veri tabanına ve veri tabanından kullanıcıya taşınır. Bu aktarım sürecinde yetkisiz kullanıcıların içeriği görmemesi için veriler şifrelenmelidir. Aktarım şifrelemesi bağlantı ile ilişkilidir.

2.1.5. Diskte şifreleme (Encryption at rest)

Diskte şifreleme ile bağlantı ile ilişkili olmayan ve veri tabanına bağlı ağdaki verileri saklandıkları yerde şifreleyerek koruma kastedilmektedir.

2.1.6. JMX kimlik doğrulama (JMX authentication)

JMX, İngilizce Java Monitoring Extension ifadesinin kısaltmasıdır ve Java İzleme Uzantısı anlamına gelen bir terimdir. Java Sanal Makinesine bağlı olarak kaynakların ilerlemesinin kontrolünü izlemek için

geliştirilmiş bir teknolojidir. Bu teknoloji, Java programlama dili kullanılarak oluşturulan uygulamalar için derinlemesine bir kontrol sağlar.

2.1.7. Kimlik doğrulama önbelleği (Authentication caching)

Önbellek, yalnızca önemli bilgiler için sınırlı depolama alanıyla isteklere hızlı yanıt için verileri depolayan geçici depolama alanıdır. Önbellekte, kullanıcı giriş bilgileri ve ilgili kullanıcıların rollerinin kapsamı ile ilgili bilgiler saklanmaktadır. Önbellekleme, kullanıcının veri tabanındaki çalışmaları sırasında yetkilendirme işlemlerini birçok kez yapma gereği duymadan, birçok oturum açma işlemini bir seferde sınırlandırır. Böylelikle kullanıcı giriş bilgileri her istekle birlikte gönderilmemiş olur ve bu da güvenliği artırıcı bir katkı sağlar. Kullanıcı rolünün önbellekte kalması için varsayılan değer milisaniye ile ölçülür ve roller önbellekte varsayılan olarak 120000 milisaniye (2 dakika) kalır. Güncelleme için aynı zaman gereklidir. Bir kullanıcının bu işlemleri yapma izni 120000 milisaniye ile sınırlıdır ve izin yetkisi 2000 milisaniyede bir güncellenir.

2.1.8. Proxy rolleri (Proxy roles)

Proxy kuralları, kullanıcının bir veri tabanı sunucusuna giriş yapmasına ve uygulama katmanında sorgu komutlarını çalıştırmasına izin verir. Proxy kuralları özellikle web sunucuları gibi güvenli ara katman yazılımları için kullanışlıdır; web sunucusu bir kez oturum açabilir ve proxy, istemci olarak sorguları yürütür ve denetim günlüğünü eksiksiz tutar. Sonuç olarak, uygulama katmanına yetki vererek kullanıcıya yetki verilmesini sınırlandırmaya yardımcı olur ve dolayısıyla kullanıcı kaynaklı olası tehditleri azaltarak verilerin güvenliğinin sağlanmasına katkı sağlar.

2.1.9. Düğümden düğüme şifreleme (Node-to-node encryption)

Düğümden düğüme şifreleme, kümedeki veri tabanı sunucuları arasındaki dahili bağlantıları SSL ile korur. Düğümden düğüme şifreleme isteğe bağlıdır ve istemciden düğüme şifrelemeden ayrıdır. Bu özellikle, bir kümedeki düğümler arasında aktarılan veriler korunabilir.

2.1.10. İstemciden düğüme şifreleme (Client-to-node encryption)

İstemciden düğüme şifreleme, kümedeki verileri şifreleme anahtarı olmayan kullanıcıların erişimine karşı korur. İstemciden düğüme şifreleme varsayılan olarak devre dışıdır ve anahtar deposundaki anahtarlarla açıkça etkinleştirilmesi gerekir.

2.2. Test Ortamı (Test Environment)

Bu çalışmada kullanılan veri tabanlarının kurulumunun yapıldığı her bir düğüm aşağıda belirtilen aynı donanım ve yazılım özelliklerine sahiptir. Küme yapılandırmasında her küme yine aynı konfigürasyona sahip dört düğümden oluşmuştur.

2.2.1. Donanım ve yazılım özellikleri (Hardware and software specifications)

Test ortamında aşağıdaki konfigürasyona sahip sunucular kullanılmıştır.

- 4 GB RAM
- Intel Core i3 işlemci, 3.33 GHz işlemci hızı
- Her birimde 200 GB geçici depolama

Test ortamında sunucular üzerine aşağıdaki yazılımlar kurulmuştur.

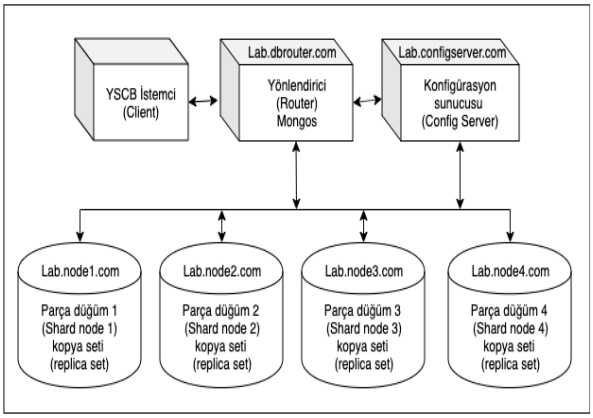
- Ubuntu 16.04 LTS (64 bit): Bu çalışma için Ubuntu 16.04 LTS uzun süreli desteklenen Linux tabanlı işletim sistemi kullanılmıştır.
- Yahoo Cloud Hizmet Göstergesi (YCSB): Yahoo laboratuvarları tarafından geliştirilen ve HBase, OrientDB, Redis, MongoDB, Cassandra ve Hypertable gibi birçok veri tabanının performansını ölçmek için kullanılan açık kaynaklı bir araçtır [13]. Bu araç ilk olarak 2010 yılında Brian F. Cooper tarafından bulut tabanlı hizmetlerin test edilmesine yönelik bir araç olarak tanıtılmış ve daha sonra birkaç SQL ve NoSQL veri tabanını içerecek şekilde geliştirilmiştir. YSCB genellikle incelenen veri tabanında okuma, yazma, güncelleme, sorgulama işlem performanslarını test etmek için kullanılır. Önceden tanımlanmış ve varsayılan olarak ayarlanmış altı farklı iş yükü olmasına rağmen, her iş yükü kullanıcının ihtiyaç duyduğu her sınamaya uygun şekilde yeniden tasarlanabilir. Örneğin, YCSB aracı tarafından gerçekleştirilen okuma ve yazma yüzdesi, kayıt büyüklüğünü ve işlem sayımlarını seçme yeteneği ile yeniden yapılandırılabilir.
- MongoDB NoSQL Veri tabanı: MongoDB NoSQL veri tabanı bu çalışmada test edilen iki veri tabanından biridir. Tek düğüm kurulumu ve yapılandırması oldukça kolaydır. Ancak, küme yapılandırması ağa ve küme düğümlerine daha fazla ayar gerektirir. Bu

çalışma için MongoDB 3.6.3 sürümünün hem kısıtlı özelliklere sahip ücretsiz lisanslı sürümü hem de gelişmiş özelliklere sahip kurumsal lisanslı sürümü kullanılmıştır [19]. MongoDB kurumsal sürümüne Ubuntu paket yöneticisi aracılığıyla da ulaşılabilir.

- Cassandra Veritabanı: Apache tarafından lisanslanan Cassandra, bu çalışmada test edilen ikinci veri tabanıdır. Bu çalışmada Java 8 ile birlikte Apache Cassandra'nın 3.11.1 sürümünün hem kısıtlı özelliklere sahip ücretsiz lisanslı sürümü hem de gelişmiş özelliklere sahip kurumsal lisanslı sürümü kullanılmıştır [20]. Cassandra kurumsal sürümü DataStax tarafından sağlanmakta ve geliştirilmektedir. Bu nedenle, artık Apache açık kaynak lisansı altında mevcut değildir. Kurulum sadece ilgili paketi DataStax web sitesinden indirerek yapılır [21].

2.2.2. MongoDB küme tasarımı (MongoDB cluster design)

MongoDB kümesinin performansını test etmek için dört MongoDB düğümü içeren bir küme yapılandırılmıştır. Küme, dört düğüm, bir DB yönlendirici ve bir konfigürasyon sunucusundan (ConfigSrv) oluşmaktadır. Şekil 2, bu çalışmada kullanılan MongoDB küme düzenine genel bir bakış sunmaktadır.



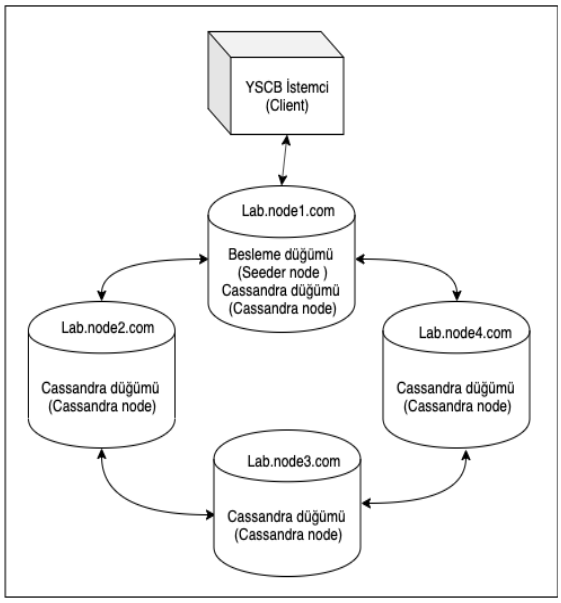
Şekil 2. MongoDB küme düzenine genel bir bakış
(MongoDB Cluster Design Overview)

2.2.3. Cassandra küme tasarımı (Cassandra cluster design)

MongoDB yaklaşımına kıyasla Cassandra veri tabanı kümesi oluşturmak ve yapılandırmak daha hızlı ve kolaydır. Bununla birlikte, MongoDB kümeleme yaklaşımı, kümenin yatay ve dikey ölçeklenebilirliğini garanti ederken, Cassandra yaklaşımı sadece yatay ölçeklendirmek için tasarlanmıştır. Bir Cassandra düğümünün, MongoDB düğümüne kıyasla konfigürasyon sunucusu (ConfigSrv.) ve DB Yönlendirici/Router işlevi gördüğüne dikkat etmek önemlidir. Şekil 3, bu çalışmada kullanılan Cassandra küme düzenine genel bir bakış sunmaktadır.

2.3. NoSQL Veri Tabanı Güvenlik Ölçütleri Test Süreci (NoSQL Database Security Criteria Testing Process)

Bu çalışmanın amacına ulaşmak için Cassandra ve MongoDB NoSQL veri tabanlarının ücretsiz açık kaynak ve kurumsal sürümlerinin, Bölüm 2.2'de açıklanan güvenlik ölçütlerini hangi seviyede desteklediklerini incelenmiştir. Bu süreçte sırasıyla tüm ölçütler her bir sürüm için incelenmiş ve sonuçlar kaydedilmiştir. Çalışmada kullanılan NoSQL veri tabanlarının güvenlik özellikleri ve dokümantasyonuna ilgili veri tabanlarının resmi web sayfalarından ulaşılmış ve bu özelliklerin mevcudiyeti test ortamında kontrol edilmiştir [22][23][24]. Buna ek olarak, her iki NoSQL veri tabanının şifreleme performans kıyaslaması, farklı YCSB iş yükleri kullanılarak yapılmıştır. İki veri tabanının şifreleme performansı aşağıdaki dört farklı iş yüküne göre test edilmiştir.



Şekil 3. Cassandra küme düzenine genel bir bakış
(Cassandra Cluster Design Overview)

1) İş Yükü A: Bu iş yükü %50 okuma (read) ve %50 yazma (write) görevi içerir.
2) İş Yükü B: Bu iş yükü %95 okuma (read) ve %5 yazma (write) görevi içerir.
3) İş Yükü E: Bu iş yükü %95 tarama (scan) ve %5 ekleme (insert) görevi içerir. Bu iş yükünde, tek kayıtlar yerine kısa kayıt aralıkları sorgulanır.
4) İş Yükü F: Bu iş yükü %50 okuma (read) ve %50 okuma-değiştirme-yazma (read-update-write) görevi içerir.

Tablo 2. Güvenlik özellikleri karşılaştırma özeti (Summary of security features comparison)

Ölçüt	Veri tabanı			
	MongoDB (ücretsiz sürüm)	Cassandra (ücretsiz sürüm)	MongoDB (kurumsal)	Cassandra (kurumsal)
Kimlik Doğrulama	✓ SCRAM, MongoDB-CR, x.509	✓ Yerleşik (Internal)	✓ SCRAM, MONGODB-CR, x.509, FIPS, Kerberos, LDAP sunucu SASL bağlantı.	✓ Yerleşik (Internal), LDAP, Kerberos
Yetkilendirme	✓ RBAC	✓ Cassandra Rol Yöneticisi	✓ RBAC	✓ RBAC, LDAP, Kerberos
Aktarım Şifreleme	✓ TLS/SSL	✓ TLS/SSL	✓ FIPS, TLS/SSL	✓ AES – DES – DESede – Blowfish / CBC – ECB / PKCS5Padding
Diskte Şifreleme	X	X	✓ OpenSSL üzerinden AES256-CBC, AES256-GCM	✓ AES – DES – DESede – Blowfish / CBC – ECB / PKCS5Padding
Denetleme	X	X	✓ kimlik doğrulama ve yetki kaydı, CRUD işlemleri kaydı kullanıcıların sistem etkinlikleri kaydı	✓ kimlik doğrulama ve yetki kaydı, CRUD işlemleri kaydı kullanıcıların sistem etkinlikleri kaydı
JMX Kimlik Doğrulama	X	✓	X	✓
Kimlik Doğrulama Önbelleği	X	✓	X	✓
Proxy Rolleri	X	X	X	✓
Düğümden Düğüme Şifreleme	X	✓	✓	✓
İstemciden Düğüme Şifreleme	✓	✓	✓	✓
Toplam desteklenen güvenlik özelliği sayısı	4	7	7	10

Testler, her bir iş yükünün sırasıyla 200K, 400K, 600K, 800K ve 1000K'lık farklı kayıt sayıları ile gerçekleştirilmiş ve elde edilen sonuçların aritmetik ortalaması alınarak her bir iş yükünün ortalama işlem süresi hesaplanmıştır.

III. SONUÇLAR VE TARTIŞMA (RESULTS AND DISCUSSIONS)

Bu bölümde sonuçlar güvenlik özelliklerinin karşılaştırmalı inceleme sonuçları ve diskte şifreleme karşılaştırma sonuçları başlıkları altında sunulmaktadır.

3.1. Güvenlik Özelliklerinin Karşılaştırmalı İnceleme Sonuçları (Results of Comparative Analysis of Security Features)

Bölüm 2.1'de tanımlanan seçili güvenlik özellikleri MongoDB ve Cassandra'nın hem açık kaynak ücretsiz hem de kurumsal sürümleri üzerinde incelendiğinde elde edilen sonuçlar Tablo 2'de özetlenmektedir.

Sonuçlar incelendiğinde hem MongoDB hem de Cassandra veri tabanının açık kaynak ücretsiz sürümlerinin kurumsal sürümleriyle karşılaştırıldığında daha az güvenlik özelliğini desteklediği görülmektedir. Ayrıca, MongoDB kurumsal sürümüyle karşılaştırıldığında Cassandra'nın kurumsal sürümünün daha fazla algoritmayı ve güvenlik özelliğini desteklediği ortaya çıkmıştır.

Güvenlik özellikleri incelemesi sonunda ortaya çıkan başlıca bulgular aşağıda listelenmiştir:

1) MongoDB (açık kaynak kodlu ve kurumsal sürüm) daha fazla kimlik doğrulama algoritmasını desteklerken, Cassandra kurumsal sürümü yetkilendirme özelliğinde öne çıkmaktadır.
2) Cassandra kurumsal sürümü, 40, 56, 112, 128, 168, 192 bit gibi çeşitli anahtar uzunluklarını seçme kabiliyeti ile verileri ağ üzerinde ve diskte şifrelemek için 256 ve 448 bit kullanan çok çeşitli şifreleme algoritmalarını desteklemektedir.
3) Cassandra (açık kaynak ve kurumsal sürüm), JMX kimlik doğrulama ve kimlik doğrulama önbelleği özelliklerini desteklemektedir. Ancak, proxy rolleri yalnızca Cassandra kurumsal sürüm tarafından desteklenirken, MongoDB bu özellikler için yerel desteğe sahip değildir.
4) MongoDB ve Cassandra açık kaynak ücretsiz sürümleri yerel veri tabanı denetim özelliği sunmazken, her iki veri tabanının kurumsal sürümleri gerekli olan iyi denetim özelliklerini sunmaktadır.
5) Düğümden düğüme ve istemciden düğüme şifrelemesi özellikleri Cassandra'nın hem açık kaynak kodlu hem de kurumsal sürümünde mevcuttur. Bu özellikler MongoDB kurumsal sürümü tarafından da desteklenmektedir. Ancak, MongoDB açık kaynaklı versiyonu yalnızca istemciden düğüme şifrelemeyi desteklemektedir.

3.2. Diskte Şifreleme Karşılaştırma Sonuçları (Encryption at Rest Benchmark Results)

Bir ağ veya disk seviyesinde verilerin şifrelenmesi, genel olarak şifreleme yükü olarak bilinir. Genel yük, gelen tüm verileri şifrelemek için şifreleme motoru tarafından harcanan süre olarak tanımlanır ve aşağıdaki formülü kullanarak hesaplanabilir:

Genel şifreleme yükü = (Çalışma zamanı (Şifreleme) - Çalışma zamanı (normal)) / Çalışma zamanı (normal)

Tablo 2'den diskte şifrelemenin sadece her iki veritabanının da kurumsal sürümünde mevcut olduğunu görebiliriz. Bu nedenle, şifreleme yük süresini hesaplamak için, YCSB aracını çalışmada kullanılan aynı iş yükleriyle veri tabanlarının kurumsal sürümleri üzerinde kullanılmış ve önce her bir veri tabanı için şifreleme olmadan geçen zamanı ölçülmüş, daha sonra şifrelemeyi etkinleştirilmiş ve aynı testleri yeniden yapılmıştır. Test sürecinde MongoDB ve Cassandra'da 256 bit anahtar uzunluğu olan AES256-CBC şifreleme algoritması kullanılmıştır. Sonuçlar Tablo 3 ve 4 ile Şekil 4 ve 5'te gösterilmektedir. Tablo 3 incelendiğinde MongoDB'de F iş yükünde şifreleme için harcanan sürenin şifresiz verileri işlemek için gereken süreden %10 fazla olduğu ve şifreleme sürecinde işlem yükünün yaklaşık %34 azaldığı görülmektedir. Bu iş yükü incelendiğinde %50 okuma ve %50 okuma-değiştirme-yazma görevi içerdiği görülmektedir. Diğer iş yüklerinde yer almayan değiştirme görevinin bu artışın sebebi olduğu söylenebilir. Bu sonuç şifreli verilerin güncelleme işlemlerinde MongoDB'nin performansının düştüğünü göstermektedir. Buna rağmen, F iş yükünde MongoDB'nin performansı Cassandra'dan yaklaşık %50 daha iyi olduğu görülmektedir (bkz. Şekil 4 ve 5).

Sonuçlar MongoDB kurumsal sürümünün genel olarak tüm çalışma yüklerinde Cassandra kurumsal sürümden ortalama %53 daha hızlı çalıştığını ve dakikada işlenen ortalama veri miktarının da %45 daha fazla olduğunu göstermektedir. Bu sonuçlarda şifrelemenin bir gereklilik olduğu ortamlarda MongoDB'nin kullanılmasının daha uygun olduğunu göstermektedir.

IV. SONUÇLAR (CONCLUSIONS)

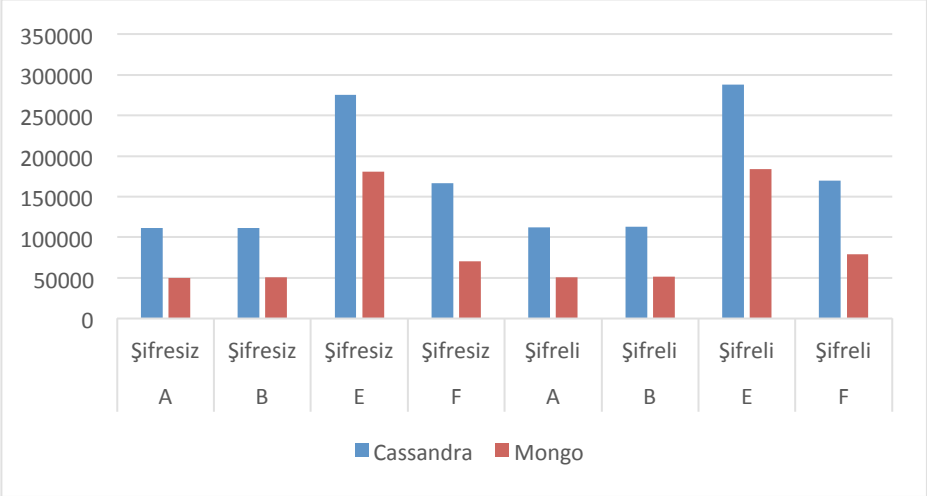
Bu çalışmada yaygın olarak kullanılan iki NoSQL veri tabanının (MongoDB ve Apache Cassandra) şifreleme performansları ve güvenlik özeliklerinin karşılaştırmalı analiz sonuçları sunulmuştur. Bu süreçte yöntem bölümünde ayrıntılı olarak anlatılan bir test ortamı kullanılmıştır. Güvenlik karşılaştırma sonuçları, her iki NoSQL veri tabanının da kayda değer güvenlik özelliklerine sahip olduğunu göstermektedir. Bununla birlikte, Cassandra'nın daha fazla güvenlik ölçütünü desteklediği ortaya çıkmıştır. Ayrıca, çalışma zamanı ve performans ile ilgili yapılan karşılaştırmada MongoDB kurumsal sürümünün şifreleme/şifre çözümleme performansının Cassandra kurumsal sürümünden ortalama %53 daha hızlı olduğu ve dakikada işleyebildiği veri miktarının ortalama %45 daha fazla olduğu bulunmuştur; bu da şifrelemenin bir gereklilik olduğu ortamlarda MongoDB'nin kullanılmasının daha uygun olduğunu göstermektedir.

Tablo 3. MongoDB şifreleme yük sonuçları (MongoDB encryption overheads results)

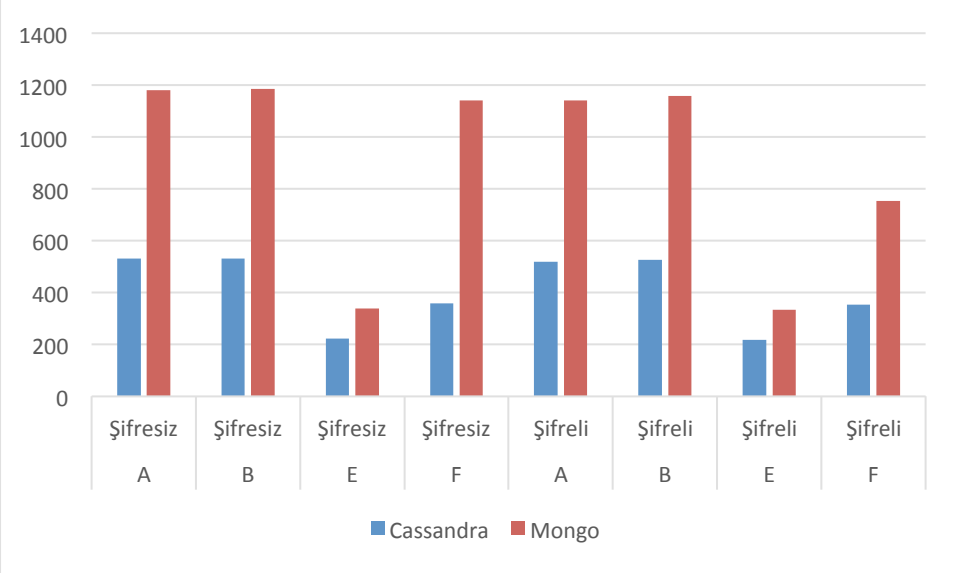
Yük (Workload)	MongoDB Enterprise Şifrelenmemiş (non-Encrypted)		MongoDB Enterprise Şifrelenmiş (Encrypted)		Hesaplanan İşlem Yükü (Overheads)	
	Çalışma süresi (sn.) (Run Time – sec.)	Dakikada işlenen ort. veri miktarı (Avg. Throughput)	Çalışma süresi (sn.) (Run Time – sec.)	Dakikada işlenen ort. veri miktarı (Avg. Throughput)	Şifreleme için harcanan fazla süre (sn.) (Time)	Şifreleme sürecinde dakikada işlenen ort. veri miktarı değişimi (%) (Throughput)
A	50027,67	1180,04	50643,01	1141,69	615,34	- 3,25
B	50555,27	1185,75	51196,53	1157,27	641,26	- 2,40
E	181008,20	340,08	183667,20	334,52	2659,00	- 1,63
F	70562,47	1140,11	79285,20	753,20	8722,73	- 33,93

Tablo 4. Cassandra şifreleme yük sonuçları (Cassandra encryption overheads results)

Yük (Workload)	Cassandra Enterprise Şifrelenmemiş (non-Encrypted)		Cassandra Enterprise Şifrelenmiş (Encrypted)		Hesaplanan İşlem Yükü (Overheads)	
	Çalışma süresi (sn.) (Run Time – sec.)	Dakikada işlenen ort. veri miktarı (Avg. Throughput)	Çalışma süresi (sn.) (Run Time – sec.)	Dakikada işlenen ort. veri miktarı (Avg. Throughput)	Şifreleme için harcanan fazla süre (sn.) (Time)	Şifreleme sürecinde dakikada işlenen ort. veri miktarı değişimi (%) (Throughput)
A	111273,40	530,39	112484,33	518,96	1210,93	- 2,15
B	111650,40	532,08	112734,13	527,08	1083,73	- 0,93
E	275627,10	223,22	288188,73	217,89	12561,63	- 2,38
F	166336,73	359,05	169537,10	353,92	3200,37	- 1,42



Şekil 4. Cassandra ve MongoDB'nin iş yüklerini tamamlama sürelerinin karşılaştırılması (Comparison of the workload completion times of Cassandra and MongoDB)



Şekil 5. Cassandra ve MongoDB'nin dakikada işlenen ortalama veri miktarlarının karşılaştırılması (Comparison of the average amount of data processed per minute by Cassandra and MongoDB)

Bellek içi veri yapısı kullanması nedeniyle gerçekleştirdiğimiz çoklu düğüm testimizde Cassandra veri tabanı performansının tüm iş yükleri için daha iyi olmasını bekliyorduk. Ancak, MongoDB'nin 3.6.3 sürümünde kullandığı yeni hafıza içi mimarisinin, MongoDB'nin önceki sürümünün test edildiği Kumar ve Roseline'ın [25] çalışmalarına kıyasla performans ve ölçeklenebilirliğinin daha iyi bir duruma geldiği söylenebilir.

Her iki veri tabanı daha önce yayınlanmış çalışmalara kıyasla güvenlik özelliklerinin geliştiği ve yeni şifreleme, yetkilendirme ve yönetim özelliklerinin eklendiğini göstermiştir [8][26]. Bununla birlikte, bu çalışmada yapılan karşılaştırma, Cassandra kurumsal sürümü, güvenlik özellikleri karşılaştırmasında en önde yer almıştır. Bunun nedeni olarak bu sürümün geliştirilmesinden ticari bir firma olan DataStaX'ın sorumlu olması gösterilebilir. Son olarak, bu çalışmada kullanılan yazılımların yeni sürümleri ile gelecek olan olası değişikliklerin önceki sürümlerle karşılaştırılması da faydalı olacaktır.

KAYNAKLAR (REFERENCES)

[1] Al-Fuqaha A., Guizani M., Mohammadi M., Aledhari M. ve Ayyash M., Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications, IEEE Commun. Surv. Tutorials, vol. 17, no. 4, pp. 2347–2376, 2015.

[2] Rao, T. R. Mitra P., Bhatt R. ve Goswami A., The big data system, components, tools, and

technologies: a survey, vol.60, no.3. Springer London, 2018.

[3] List of NoSQL Database Management Systems. URL: https://nosql-database.org. 27-Kas-2019 tarihinde erişilmiştir.

[4] Abubakar Y., Adeyi T. S. ve Auta I. G., Performance Evaluation of NoSQL Systems using YCSB in a Resource Austere Environment, Int. J. Appl. Inf. Syst., vol. 7, no. 8, pp. 23–27, 2014.

[5] Klein J., Gorton I., Ernst N., Donohoe P., Pham K. ve Matser C., Performance evaluation of NoSQL databases: A case study, in PABS 2015 - Proceedings of the 1st ACM/SPEC International Workshop on Performance Analysis of Big Data Systems, 2015, pp.5–10.

[6] Abramova V., Bernardino J. ve Furtado P., Which NoSQL Database? A Performance Overview, Open J. Databases, vol.1, no.2, pp. 17–24, 2014.

[7] DB-Engines Ranking. URL: https://db-engines.com/en/ranking. 27-Kas-2019 tarihinde erişilmiştir.

[8] Okman L., Gal-Oz N., Gonen Y., Gudes E. ve Abramov J., Security issues in NoSQL databases, in Proc. 10th IEEE Int. Conf. on Trust, Security and Privacy in Computing and Communications, TrustCom 2011, 8th IEEE Int. Conf. on Embedded Software and Systems, ICESS 2011, 6th Int. Conf. on FCST 2011, 2011, pp. 541–547.

[9] Sahafizadeh E. ve Nematbakhsh M. A., A Survey on Security Issues in Big Data and NoSQL, Adv. Comput. Sci. An Int. J., vol. 4, no. 4, pp. 68–72, 2015.

[10] Chahal D., Kharb L. ve Gupta M., Challenges and Security Issues of NOsql Databases, Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol., vol. 2, no. 5, pp. 976–982, 2017.

[11] Grim M.W. ve Wiersma A.T., Security Analysis of Encrypted NoSQL Databases, University of Amsterdam, 2017.

[12] Shahriar H. ve Haddad H. M., Security Vulnerabilities of NoSQL and SQL Databases for MOOC Applications, Int. J. of Dig. Soc. (IJDS), vol. 8, no. 1, pp. 1244–1250, 2017.

[13] Cooper B. F., Silberstein A., Tam E., Ramakrishnan R. ve Sears R., Benchmarking cloud serving systems with YCSB, in Proceedings of the 1st ACM Symposium on Cloud Computing, SoCC'10, 2010, pp. 143–154.

[14] Sahafizadeh E. ve Nematbakhsh M. A., A Survey on Security Issues in Big Data and NoSQL, Int. J. Adv. Comput. Sci. vol. 4, no. 4, pp. 68–72, 2015.

[15] Srinivas S. ve Nair A., Security maturity in NoSQL databases - Are they secure enough to haul the modern IT applications?, 2015 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2015, pp. 739–744, 2015.

[16] Cuzzocrea A. ve Shahriar H., Data masking techniques for NoSQL database security: A systematic review, Proc. - 2017 IEEE Int. Conf. Big Data, Big Data 2017, vol. 2018-Janua, pp. 4467–4473, 2017.

[17] Gharajeh M. S., Security Issues and Privacy Challenges of NoSQL Databases, in NoSQL Database for Storage and Retrieval of Data in Cloud, Ganesh Chandra Deka, Ed. Boca Raton: CRC Press, 2017, pp. 271–289.

[18] Cattell R., Scalable SQL and NoSQL Data Stores, SIGMOD Rec., vol. 39, no. 4, pp. 12–27, May 2011.

[19] MongoDB. What is MongoDB? URL: http://docs.mongodb.org/manual/core/introduction/#what-is-mongodb 02-May-2019 tarihinde erişilmiştir.

[20] Apache. What is Cassandra? URL: https://cassandra.apache.org/ 02-May-2019 tarihinde erişilmiştir.

[21] DataStax. DataStax Enterprise. URL https://www.datastax.com/resources/datasheet/datastax-enterprise. 02-May-2019 tarihinde erişilmiştir.

[22] MongoDB official website. Security Checklist. URL: https://docs.mongodb.com/manual/administration/security-checklist/ 02-May-2019 tarihinde erişilmiştir.

[23] Apache Cassandra official website. Security.URL: https://cassandra.apache.org/doc/latest/operating/security.html 02-May-2019 tarihinde erişilmiştir.

[24] DataStax official website. DataStax 5.1 Security Guide. URL: https://docs.datastax.com/en/security/5.1/index.html 02-May-2019 tarihinde erişilmiştir.

[25] Kumar R. ve Mary R. R., "Comparative Performance Analysis of various NoSQL Databases : MongoDB, Cassandra and HBase on Yahoo Cloud Server," Imp. J. Interdiscip. Res., vol.3, no.4, pp. 265–269, 2017.

[26] Zahid A., Masood R. ve Shibli M. A., Security of sharded NoSQL databases: A comparative analysis, in Conference Proceedings - 2014 Conference on Information Assurance and Cyber Security, CIACS 2014, 2014, pp. 1–8.