Veb-ilovalarda sessiya va xavfsizlik nazariyasi

Samarqand davlat pedagogika instituti
Abdullayev Ilxom Xujayorovich samexpoler@gmail.com
Nurmurodova Jasmina Davronbek qizi jasminanurmurodova14@gmail.com

Annotatsiya

Ushbu ilmiy maqola veb-ilovalarda sessiya boshqaruvi va xavfsizlik masalalarini tahlil qiladi. Veb-ilovalar sessiya boshqaruvi orqali foydalanuvchilarning tizimga kirish huquqini nazorat qiladi, lekin ushbu tizimlar ko‘plab xavf-xatarlarga duch keladi. Sessiya hijacking (sessiya o‘g‘irlash) va sessiya fixating (sessiya belgilash) kabi zaifliklar kiberhujumchilarga foydalanuvchilarning hisoblariga ruxsatsiz kirishga imkon beradi. Maqolada veb-ilovalarning xavfsizligini ta'minlash uchun zarur bo‘lgan xavfsizlik choralari, jumladan, HTTPS protokoli, murakkab sessiya ID-lari va sessiya vaqtini cheklash kabi usullar keltirilgan.

Shuningdek, maqolada sessiya xavfsizligini o‘lchash va xavfsizlikni yuqori miqyosda baholash uchun zamonaviy testlar, metodlar va tavsiyalar berilgan. Veb-ilovalarda cookie va sessiya xavfsizligi, shuningdek, ularni o‘g‘irlash xavfini kamaytirish uchun shifrlash texnologiyalari va xavfsizlik protokollari muhokama qilinadi. Maqola sessiya boshqaruvini yaxshilash va veb-ilovalar xavfsizligini oshirish uchun muhim tavsiyalarni taqdim etadi.

Kalit so‘zlar: *veb-ilova, sessiya boshqaruvi, xavfsizlik, sessiya hijacking, HTTPS protokoli.*

Sessiya boshqaruvi zamonaviy veb-ilovalarning muhim tarkibiy qismi hisoblanadi, chunki u serverga bir nechta so‘rovlar orasida foydalanuvchiga xos holatni, masalan, autentifikatsiya qilingan hisobni kuzatishga imkon beradi. Sessiya boshqaruvi hujumchilar uchun jozibali va yuqori qiymatga ega maqsad bo‘lib, bu OWASP (Open Web Application Security Project) tomonidan tuzilgan veb-ilovalar xavfsizligi xatarlarining eng yirik 10 taligida uchinchi o‘rinni egallaydi. Sessiya boshqaruviga nisbatan eng keng tarqalgan hujumlar sessiya hijacking (sessiya o‘g‘irlash) [13], sessiya fixating (sessiya belgilash) yoki tarmoq orqali tinglash hujumlaridir. Yaqinda chop etilgan ish [5] Alexa reytingidagi eng yaxshi 100 saytning 10 tasining sessiya o‘g‘irlashga qarshi zaif ekanligini aniqladi. Jamoaviy tarmoqlar va hotspotlarning ko‘payishi bilan, autentifikatsiya qilingan sessiyalarni o‘g‘irlash oddiy tugma bosish orqali amalga oshirilishi mumkin, bu esa Firesheep [4] kabi vositalar orqali ko‘rsatilgan.

Sessiya boshqaruvidagi bunday yovuz hujumlarning oldini olish uchun, amaldagi xavfsizlik bo‘yicha ko‘rsatmalar TLS (Transport Layer Security) protokolini qo‘llashni va Secure hamda HttpOnly cookie atributlarini ishlatishni tavsiya qiladi. TLS aloqa kanalini tinglash yoki tarmoq manipulyatsiyasi hujumlaridan himoya qiladi, cookie atributlari esa sessiya hujumlaridan foydalanuvchi brauzeri ichidagi zararli kodlardan himoya qiladi. Garchi bu xavfsizlik ko‘rsatmalari sessiya boshqaruvidagi hujumlarga qarshi etarli darajada himoya taqdim etsa ham, va shuning uchun ularni amal qilish zarur bo‘lsa-da, TLS internetda keng tarqalish tezligi pastligiga olib kelmoqda: 2010-yilda Qualys tomonidan o‘tkazilgan tadqiqotga ko‘ra, 119 million domenning faqat 0.72 millioni o‘z domen nomi bilan TLS sertifikatini taqdim etadi [15], shundan ko‘p qismining (taxminan 25%) to‘g‘ri tekshirilmasligi aniqlangan. Sezilarli darajada sekinlashgan

qabul qilish sur'atining sabablari aniq belgilangan emas, ammo adabiyotlar bir qancha imkoniyatlarni taqdim etadi [1, 5, 6, 12].

HTTP (yoki TLS bo‘lmagan ilovalar) va TLS (HTTPS) ilovalari birgalikda mavjud bo‘lishi ehtimoli davom etishi sababli, HTTP ilovalarida sessiya boshqaruvining xavfsizligini oshirish juda tavsiya etiladi. Ushbu maqolada, HTTP protokoli asosida joylashgan veb-ilovalarda sessiya hijacking, sessiya fixating yoki tarmoq orqali tinglash hujumlariga qarshi himoya qilish uchun xavfsiz va engil vaznli sessiya boshqaruvi mexanizmini taklif etamiz. Bizning sessiya boshqaruvi mexanizmimiz faqat sessiya identifikatoriga tayanmasdan, brauzer va server o‘rtasida umumiy sessiya maxfiyligini o‘rnatadi va uni brauzerda xavfsiz tarzda saqlaydi. Umumiy maxfiylik web sahifalariga ochiq bo‘lmaydi va tarmoq orqali uzatilmaydi. Aniqroq qilib aytganda, bizning xavfsiz sessiya boshqaruvi mexanizmimiz quyidagi himoya imkoniyatlarini taqdim etadi: (a) Script yoki DOM asosidagi sessiya hijacking yoki sessiya fixating hujumlaridan himoya, (b) Tarmoq orqali tinglash hujumlaridan himoya, (c) Yangi va eski veb-ilovalarga shaffof tarzda joriy etish, infrastrukturaga asoslangan tarzda saqlash.

Hozirgi kunda amalda qo‘llaniladigan sessiya boshqaruvi mexanizmi sessiya identifikatorini saqlash uchun cookies-dan foydalanadi, bu esa to‘g‘ridan-to‘g‘ri serverda saqlanadigan holatga bog‘langan. Server holati odatda sessiya haqidagi ma’lumotlarni, masalan, foydalanuvchining shaxsiyati va autentifikatsiya holatini saqlash uchun ishlatiladi. Server sessiya identifikatorini Set-Cookie sarlavhasi orqali chiqaradi va brauzer bu identifikatorni har bir so‘rovda Cookie sarlavhasi orqali maqsadli domenga uzatadi. Bunday holatda, sessiya identifikatori bearer token (taqdim etuvchi token) bo‘ladi [6], ya’ni kimki ushbu tokenni serverga taqdim etsa, server tomonidagi holatga kirish huquqini oladi.

Sessiya boshqaruvi mexanizmlariga nisbatan hujumlar sessiya identifikatorining bearer token sifatidagi moslashuvchanligini suiiste’mol qiladi. Tarmoqdagi trafikni tinglovchi odam har qanday xavfsiz bo‘lmagan aloqada sessiya identifikatorini o‘g‘irlashi mumkin, bu esa Firesheep [4] brauzer qo‘shimchasi orqali bir nuqtada va bitta bosish orqali sodir bo‘ladi. Bundan tashqari, tarmoqdagi zararli reklama kiritish orqali maqsadli sayt kontekstida skriptlarni ishlata oladigan dushmanlar, cookie identifikatorlarini o‘qish va yozish orqali sessiya identifikatorlarini o‘zgartirishi mumkin. Sessiya identifikatorini o‘qish sessiya o‘g‘irlashga olib kelishi mumkin, va uni yozish sessiya belgilashga olib keladi, bu ikkala hujum ham hujumchiga foydalanuvchining autentifikatsiya qilingan sessiyasiga to‘liq nazorat qilish imkoniyatini beradi.

Hozirgi xavfsizlik bo‘yicha ko‘rsatmalar sessiya boshqaruvi mexanizmlariga qarshi yaxshi ma’lum bo‘lgan hujumlarga qarshi himoya qilishni o‘z ichiga oladi. document.cookie xususiyati orqali sessiya cookie’larini manipulyatsiya qilishga asoslangan hujumlar HttpOnly bayrog‘i yordamida oldini olish mumkin, bu esa JavaScript tomonidan cookie’larga kirishni taqiqlaydi. So‘nggi vaqtlarda HttpOnly bayrog‘ining qabul qilinishi nisbatan cheklangan edi [13], ammo hozirgi kunda bir qancha ramkalar avtomatik ravishda sessiya cookie’lariga HttpOnly qo‘shmoqda [2, 18].

Tinglash hujumlarini bartaraf etish uchun barcha aloqa xavfsiz kanal orqali amalga oshirilishi kerak. Veb-ilovalar odatda HTTPS (TLS bilan himoyalangan HTTP)dan foydalanadi. Biroq, cookie'lar xavfsiz va xavfsiz bo'lmagan aloqalar orqali almashiladigan bo'lgani uchun ularga maxsus e'tibor berilishi kerak. Sessiya identifikatorini o'z ichiga olgan cookie'ga Secure bayrog'ini qo'shish uning maxfiyligini kafolatlaydi, chunki brauzer uni xavfsiz bo'lmagan aloqada yubormaydi.

Bu xavfsizlik choralari sessiya boshqaruviga qarshi eng keng tarqalgan hujumlardan himoya qilish uchun etarlicha himoya ta'minlaydi, ammo u veb-ilovalar murakkabligini va veb-ishlab chiquvchilarga yuklamani oshiradi. Ushbu xavfsizlik ko'rsatmalaridan birortasini bajarishdagi xato, veb-ilova sessiya boshqaruvi mexanizmiga qarshi bir nechta hujumlarni darhol amalga oshirishi mumkin.

Sessiya boshqaruvini xavfsizlashtirish - bu faol tadqiqot sohasidir, va bu muammoni hal qilish uchun bir nechta turli yondashuvlar ishlab chiqilgan. Mana shu yondashuvlardan eng mos akademik ishlarni taqdim etamiz. Bu takliflarning barchasi bir nuqtada TLS yordamida credential (akkreditatsiya) almashish yoki umumiy maxfiylikni o'rnatishni o'z ichiga oladi. Shuningdek, ba'zi yechimlar, ular joriy qilinishi uchun, merosga olingan ilovalarda katta o'zgarishlarni talab qiladi yoki autentifikatsiya jarayoniga qattiq integratsiyalashgan. Bu cheklovlar ularni bir nechta umumiy veb-ssenariylar bilan mos kelmaydi, masalan, eski ilovalarni yoki uchinchi tomon autentifikatsiya xizmatlarini qo'llab-quvvatlashda. Shunga qaramay, bu yechimlarning har biri o'zining afzalliklariga ega va xavfsiz sessiya boshqaruvi bo'yicha qimmatli fikrlarni taqdim etadi.

SessionLock [1] HMAC asosida, TLS kanali orqali o'rnatilgan umumiy sessiya maxfiyligiga asoslangan so'rovlarni qo'shimcha qiladi. Sessiya maxfiyligi xavfsiz cookie ichida saqlanadi va HTTP sahifalari uchun URL ning fragment identifikatorida saqlanadi, bu esa tarmoq orqali hech qachon yuborilmaydi. SessionLock shuningdek, Diffie-Hellman yordamida TLS bo'lmagan kanal orqali o'rnatishni qo'llab-quvvatlaydi. Biroq, bu yondashuvning kamchiligi shundaki, u JavaScript kutubxonasi orqali amalga oshirilgan va injeksiya hujumlariga qarshi zaif. Bundan tashqari, SessionLock barcha so'rovlarni AJAX orqali JavaScript orqali amalga oshirishni talab qiladi, bu esa ko'pgina merosga olingan ilovalar bilan mos kelmaydi.

BetterAuth [11] veb-ilovalar uchun autentifikatsiya protokoli bo'lib, tarmoq hujumlari, phishing va cross-site request forgery hujumlariga qarshi himoya taqdim etadi. BetterAuth foydalanuvchining parolini umumiy maxfiylik sifatida hisoblaydi va bu maxfiylikni bexavfsiz kanalda sessiya maxfiyligini kelishish uchun ishlatadi.

One-Time Cookies [5] sessiya identifikatorini statik o'rniga har bir so'rov uchun bir martalik tokenlar bilan almashtirishni taklif qiladi. Har bir token faqat bir marta ishlatilishi mumkin, lekin boshlang'ich umumiy maxfiylik yordamida har bir token alohida tekshiriladi va mavjud sessiyaga bog'lanadi.

TLS Origin-Bound Certificates [6] esa TLS uchun kengaytma bo'lib, brauzer va server o'rtasida kuchli autentifikatsiya kanali o'rnatadi. Bu xavfsiz kanal ichida TLS-OBC cookie'lar va

uchinchi tomon autentifikatsiya tokenlarini bog‘lashni qo‘llab-quvvatlaydi, bu esa bearer tokenlar o‘g‘irlanishini oldini oladi.

HTTP Integrity Header [7] esa HTTP-ga yaxlitlikni himoya qilishni taklif etadi. Bu sarlavha TLS yoki an’anaviy Diffie-Hellman almashinuvi orqali kalit almashishdan so‘ng, xabarni tanlangan qismlarining yaxlitligini himoya qiladi.

Sessiya boshqaruvi zamonaviy veb-ilovalarning muhim tarkibiy qismi hisoblanadi, ammo ko‘pincha tarmoq yoki skript asosidagi hujumlarga duch keladi. Hozirgi kunda qo‘llaniladigan de-fakto yechim TLSni o‘rnatishdir, lekin ko‘plab ilovalar turli sabablarga ko‘ra, buni to‘g‘ri amalga oshirmaydi. Biz TLSni talab qilmaydigan, ammo tinglash, sessiya o‘g‘irlash va sessiya belgilash hujumlariga qarshi himoya qiladigan xavfsiz va engil vaznli sessiya boshqaruvi mexanizmini taqdim etdik. Bizning sessiya boshqaruvi mexanizmimizni uchinchi tomon autentifikatsiya xizmatlarini o‘z ichiga olgan odatiy TLSsiz senariyada qanday joriy qilish mumkinligini ko‘rsatdik. Shuningdek, xavfsiz sessiya boshqaruvining vebda qanday joriy qilinishi va bizning yechimimizni nega engil deb hisoblashimizni tushuntirdik.

Foydalanilgan adabiyotlar

1. B. Adida. Sessionlock: securing web sessions against eavesdropping. In Proceedings of the 17th international conference on World Wide Web, pages 517–524, 2008.
2. Apache Software Foundation. Apache tomcat - migration guide - tomcat 7.0.x. Online at http: // tomcat. apache. org/ migration-7. html , 2012.
3. A. Barth, C. Jackson, and J. C. Mitchell. Robust defenses for cross-site request forgery. In Proceedings of the 15th ACM conference on Computer and communications security, pages 75–88, 2008.
4. E. Butler. Firesheep. Online at http: // codebutler. com/ firesheep , 2010.
5. I. Dacosta, S. Chakradeo, M. Ahamad, and P. Traynor. One-time cookies: Preventing session hijacking attacks with stateless authentication tokens. ACM Transactions on Internet Technology (TOIT), 12(1):1, 2012.
6. M. Dietz, A. Czeskis, D. Balfanz, and D. S. Wallach. Origin-Bound Certificates : A Fresh Approach to Strong Client Authentication for the Web. In Proc. 21st USENIX Security Symposium, 2012.
7. P. Hallam-Baker. Http integrity header. Online at http: // tools. ietf. org/ html/ draft-hallambaker-httpintegrity-02 , 2012.
8. E. Hughes. An encrypted key transmission protocol. rump session of CRYPTO, 94, 1994.
9. G. Inc. Federated login for google account users. Online at https: // developers. google. com/ accounts/ docs/ OpenID , 2013.
10. C. Jackson and A. Barth. Forcehttps: Protecting High-SecurityWeb Sites from Network Attacks. In Proceeding of the 17th international conference on World Wide Web, pages 525—-534, Apr. 2008.

11. M. Johns, S. Lekies, B. Braun, and B. Flesch. BetterAuth: Web Authentication Revisited. In Proceedings of the 28th Annual Computer Security Applications Conference, pages 169—-178, Dec. 2012.
12. A. Langley, N. Modadugu, and W. Chang. Overclocking ssl. In Velocity: Web Performance and Operations Conference, 2010.
13. N. Nikiforakis, W. Meert, Y. Younan, M. Johns, and W. Joosen. Sessionshield: lightweight protection against session hijacking. Engineering Secure Software and Systems, pages 87–100, 2011.
14. D. Recordon and B. Fitzpatrick. OpenID authentication 2.0. pages 1–35, 2007.
15. I. Ristic. Internet ssl survey 2010. Talk at BlackHat, 2010