digital whisper
Technologic Papers

DNSSEC

מאת: אריק פרידמן

רקע - פרוטוקול DNS תחת התקפה

פרוטוקול DNS (Domain Name System) הוא אחת מאבני-הבניין הבסיסיות ביותר של האינטרנט. זהו הפרוטוקול המאפשר תרגום של כתובות האינטרנט שאנו מזינים לדפדפן (כמו www.digitalwhisper.co.il) לכתובות המספריות בהן נעשה שימוש על-ידי פרוטוקולי התקשורת בין המחשבים (כתובות IP, כמו 193.105.99.162). אפשר לחשוב על DNS כעל ספר הטלפונים של האינטרנט. ספר הטלפונים הזה מבוזר בין מספר רב של שרתים, שרתי DNS, ופרוטוקול DNS מאפשר לפנות לשרתים אלה כדי לבצע את התרגום.

בגלל חלקו המכריע של פרוטוקול DNS בתפקוד התקין של האינטרנט, הוא הפך גם למטרה אטרקטיבית להתקפות. למעשה, כאשר האקרים למיניהם מאיימים "להשבית את האינטרנט" (כמו במקרה של אנונימוס), הם לרוב מכוונים לפגוע בתשתית של DNS, ומניפולציה של שרתי DNS היא גם אחד האמצעים בארגז הכלים של ממשלות שמנסות לשלוט בגישה לאינטרנט.

חלק לא מבוטל מההתקפות על DNS מכוונות "להזריק" רשומות DNS כוזבות לזכרון המטמון של שרתי DNS, באמצעות זיוף תשובות לשאילתות DNS כאילו הן מגיעות משרת DNS אמיתי. התקפות אלה מכונות Cache Poisoning, והן מתאפשרות בעיקר כיוון שהפרוטוקול מתייחס לכל תשובת DNS ש"מתאימה" לשאילתת DNS שנשלחה, כתשובה אמיתית. ההגדרה של מהי תשובת DNS "מתאימה" השתנתה לאורך הזמן, כאשר נקודות תורפה שונות בפרוטוקול נוצלו כדי לבצע התקפות, וגרסאות חדשות של שרתי DNS העלו את רף בדיקות ההתאמה כדי לסתום את הפרצות. למשל, ב-2008 חוקר אבטחת המידע דן קמינסקי חשף נקודת תורפה כזו בהתקפה שכיוונה לזייף רשומות המצביעות לשרתי ה-DNS עצמם. כל שאילתה שנשלחת מכילה מספר מזהה (queryID), שצריך להימצא גם בתשובה, והתוקף יכול לקלוע למספר המזהה הנכון על-ידי שליחת מספר רב של תשובות עם ניחושים, אפילו אם המספר המזהה נבחר באקראי. הפתרון שקמינסקי הציע היה לוודא כי גם מספר הפורט המשמש לשליחת השאילתות הינו אקראי, כך שהתוקף יצטרך לנחש גם אותו. פתרון זה שימש כדרך סבירה להקטין משמעותית את הסתברות ההצלחה של ההתקפה, ולהפוך אותה ללא מעשית. עם זאת, זהו בגדר "פלסטר" המספק פתרון לפרוטוקול שאינו בטוח. פרוטוקול DNSSEC מנסה לפתור בעיות מסוג זה מהיסוד, באמצעות שילוב תהליכי אימות קריפטוגרפיים בפרוטוקול.

המטרה המרכזית של הפרוטוקול היא לספק אימות (authentication) כחלק מהפרוטוקול, כדי לוודא שתשובות DNS נשלחות משרת לגיטימי, וכן שלמות (integrity) של ההודעות, כלומר, וידוא שאף גורם זדוני לא שינה הודעות בדרך. הפרוטוקול אינו מיועד לספק סודיות, כך שכמו ב-DNS רגיל, התוכן של DNSSEC אינו מוצפן וכל אחד יכול לקרוא אותו.

ההיסטוריה של DNSSEC

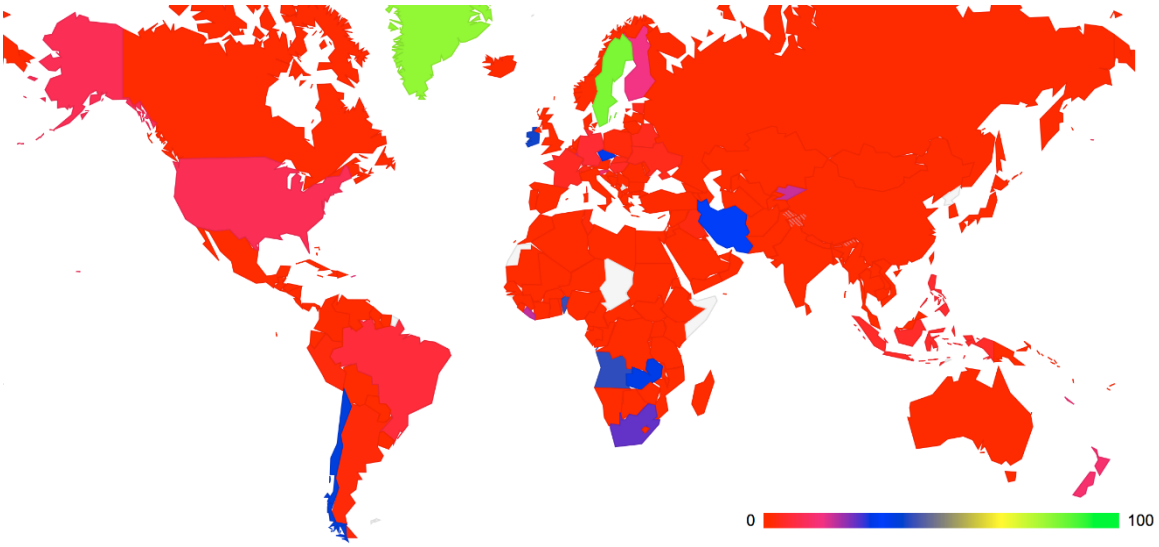
פרוטוקול DNSSEC פותח במסגרת ארגון IETF (Internet Engineering Task Force), ארגון בין-לאומי שאחראי לפעילות תקינה של האינטרנט, ובפרט לקביעת התקנים שבבסיס רשת האינטרנט. DNSSEC הפך לנושא בטיפול IETF ב-1994, כאשר אחד הגורמים המאיצים לפעילות היה פרסום מאמר של סטיבן בלובין על החולשות של DNS (המאמר נכתב עוד ב-1990, אך פורסם רק ב-1995).

ב-1997 קבוצת העבודה של IETF פרסמה את התקן הראשון, RFC2065. לאחר כשנתיים פורסמה גרסה מתוקנת, RFC2535, בעקבות משובים מהמפתחים הראשונים, ותוכנת BIND9 הייתה המימוש הראשון של שרת DNS שתמך ב-DNSSEC. עם זאת, הפתרון הראשוני לא היה מוצלח, בעיקר כיוון שלא היה מתאים לפריסה בהיקף רחב. הלקחים נלמדו וב-2005 פורסמה סדרת תקנים משוכתבת, RFC 4033-4035. זמן קצר לאחר-מכן, באוקטובר 2005, שוודיה (.SE) הייתה שרת האינטרנט הארצי הראשון שפרס DNSSEC. למרות שהתקן החדש פתר רבות מהבעיות של התקן המקורי, גלגלי האינטרנט טוחנים לאט. רק ביולי 2010 ארגון ICANN (Internet Corporation for Assigned Names and Numbers) פרסם מפתחות עבור שרתי השורש, שבראש היררכיית DNS. רק באפריל 2011 המתחם com נחתם על-ידי מפתחות תקפים. הפריסה של DNSSEC עדיין נמשכת - נכון לספטמבר 2012, ישנם 64 שמות מתחם ברמת מדינה (ccTLDs) החתומים עם DNSSEC. שם המתחם של ישראל, il., אינו אחד מהם.

איפה DNSSEC עומד היום

באוקטובר 2012 ג'ף הוסטון וג'ורג' מיכלסון, מדענים ב-APNIC (רשם האינטרנט האחראי על אסיה ואוקיאניה), פרסמו שתי רשומות ובהן סטטיסטיקות לגבי השימוש ב-DNSSEC נכון לספטמבר 2012. הערכתם הראשונית הייתה כי כ-4% משירותי ה-DNS היו מסוגלים לבצע אימות של DNSSEC, וכ-9% מעמדות הקצה השתמשו בשירותי DNS שהיו מסוגלים לבצע אימות כזה. לאחר בחינה זהירה ומחמירה יותר של המידע שאספו, הם עדכנו את ההערכות שלהם, והסיקו שלמעשה רק 1.7% משירותי ה-DNS מבצעים אימות DNSSEC, ורק 1.6% מעמדות הקצה משתמשות בלעדית בשירותי DNS שמאמתים רשומות DNSSEC. המדינות שנמצאו מובילות בהטמעת DNSSEC היו שוודיה (83% מהבקשות עובדו בידי

שירותי DNS התומכים ב-DNSSEC), אנגולה (41%) ואירלנד (39%). עבור ישראל, רק שירות DNS אחד מתוך 297 שנדגמו (0.34%) ביצע אימות DNSSEC.



החלק היחסי של שירותי ה-DNS בכל מדינה, שמבצעים אימות DNSSEC
מקור: http://www.potaroo.net/ispcol/2012-10/counting-dnssec-2.html

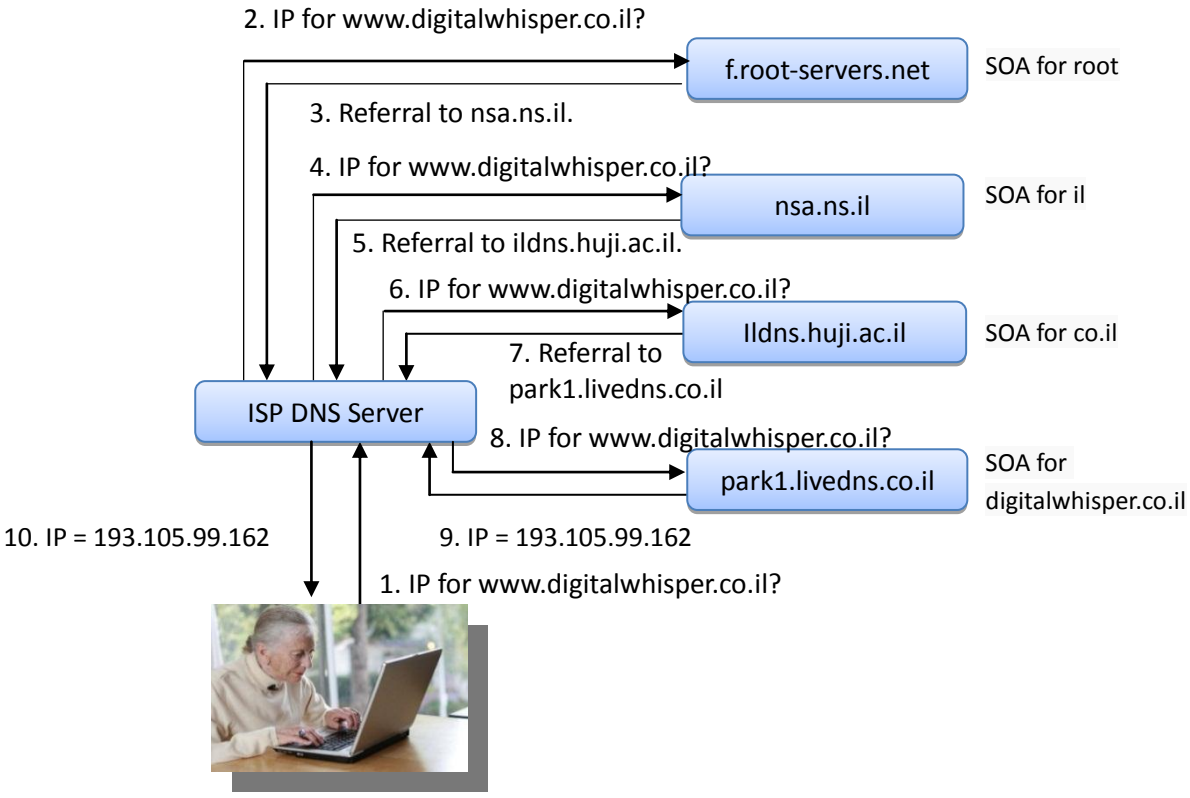
תזכורת - איך DNS עובד

תיאור מלא של פרוטוקול DNS חורג מהיקף כתבה זו, אולם כדי להסביר את העקרונות של DNSSEC, להלן תזכורת קצרה על פעולתו של פרוטוקול DNS. התיאור שלהלן חלקי ביותר, אך הוא תופס את הנקודות העיקריות בתהליך.

לכל מחשב המתחבר לרשת האינטרנט מוגדר מיהו שירות ה-DNS אליו יש לשלוח בקשות לתרגום כתובות. למשל, כאשר מתחברים לרשת האינטרנט דרך ספק אינטרנט כמו נטוויז'ן, בזק בינלאומי, או 012, בתהליך החיבור יוגדר שרת DNS מטעם ספק האינטרנט כמען לשאילתות DNS. שרת זה יקבל את השאילתות ממחשב הלקוח, יבצע עבורו את תהליך התרגום (במהלכו ישלח שאילתות לשרתים אחרים), ולבסוף יחזיר לו את התשובה.

בבסיס הפרוטוקול עומדת היררכיה של שמות מתחם - בראשה השורש; לאחר-מכן שמות המתחם הראשים (Top Level Domains) הכוללים שמות מתחם כמו com, org, edu, וגם שמות מתחם של מדינות (il, au, it, ...), המכונים גם ccTLD (country code Top Level Domains); ולאחר-מכן ההיררכיה של שמות המתחם ממשיכה להתפצל לתתי-מתחמים, כגון co.il, cnn.com, digitalwhisper.co.il, וכן הלאה. עבור כל אחד משמות המתחם מוגדר מי שרת ה-DNS האחראי עליו (authoritative server, או Source of Authority), ושרת זה הוא "הכתובת הרשמית לשאלות" עבור כל הכתובות שתחת אותו שם מתחם.

כאשר נשלחת לשירות DNS שאילתה לגבי שם מתחם מסוים, ואין בזכרון המטמון של השרת שום מידע לגבי שם זה, השרת בדרך-כלל פותח בתהליך איטרטיבי שבו הוא פונה לשרתי DNS אחרים כדי למצוא את התשובה. בהנחה שלשרת אין מידע על אף אחד מהשרתים האחרים בדרך, נקודת ההתחלה היא באחד משרתי השורש, שאת כתובתם מכירים כל שרתי ה-DNS. החל מנקודה זאת השרת יקבל הפניות לשרתים אחרים, במורד ההיררכיה של DNS. למשל, בהינתן שאילתה על www.digitalwhisper.co.il, שרת השורש יפנה את השואל אל שרת ה-DNS האחראי על il, הוא בתורו יפנה את השואל אל שרת השמות של co.il, שיפנה אותו אל שרת השמות האחראי על digitalwhisper.co.il. יש בתהליך בעיה של ביצה ותרנגולת, כי ההפניה לשרתים אחרים דורשת בעצמה לדעת את כתובות השרתים, ומכאן את תרגום הכתובות שלהם לכתובות IP. הפרוטוקול פותר את הבעיה על-ידי החזרת מידע נוסף ביחד עם ההפניה. מידע זה מועבר ברשומות המכונות "רשומות דבק" (glue records) - למשל, ביחד עם ההפנייה לשרת nsa.ns.il (אחד מהשרתים האחראים על il), התשובה משרת השורש תכיל מידע האומר "כתובת ה-IP של nsa.ns.il היא 92.115.210.58", וכך תאפשר לפנות ישירות לשרת הבא בהיררכיה. התרשים הבא מתאר תהליך מלא של תרגום הכתובת www.digitalwhisper.co.il לכתובת IP.[1]



[1] במציאות התהליך יתבצע בצורה קצת שונה - למשל, כיוון שאותם שרתי שמות משמשים גם את il וגם את co.il, יהיו פחות שלבים בתהליך התרגום. הדיוק הוקרב לטובת המחשת התהליך ההיררכי.

המידע העובר בתשובות על שאילתות DNS מסודר בקבוצות של רשומות המכונות "רשומות משאב" (Resource Records ובקיצור RR). למשל, רשומות משאב מסוג "A" הן הרשומות המכילות תרגום של כתובת אינטרנט לכתובת IP, והן הרשומות שבשימוש הנפוץ ביותר בפרוטוקול DNS. רשומות משאב מסוג "NS" (קיצור של Name Server) מדווחות מי הוא שרת ה-DNS האחראי על שם מתחם נתון. בשאילתת DNS מצוין מה הוא סוג הרשומה המבוקש (למשל A או NS), ולפי זה שרת ה-DNS יודע איזה נתון לספק בחזרה.

הרעיון המרכזי ב-DNSSEC הוא לבצע חתימות קריפטוגרפיות על רשומות המשאב. בפרט, רשומות המשאב נחתמות על-ידי הגורם האחראי עליהן. לדוגמה, רשומות הכתובת של www.digitalwhisper.co.il ייחתמו על-ידי שרת ה-DNS האחראי על שם מתחם זה, livedns.co.il. וידוא שחתימות אלה תקינות מבטיח שתוכן הרשומות לא השתנה בדרך מהשרת העונה, והשימוש במפתח הקריפטוגרפי מספק הוכחה לזהות מקור תוכן הרשומות. החתימות מצורפות לתשובת ה-DNS, וניתן לשמור אותן בזכרון המטמון של שרתי DNS ביחד עם שאר חלקי התשובה. לחתימות יש זמן תפוגה, שאחריו הן לא תקפות והשרת צריך להנפיק חתימה חדשה.

כדי לאפשר ביצוע ווידוא של חתימות, לכל שרת שמות של DNS ניתן להקצות זוג מפתחות, מפתח פרטי ומפתח פומבי. המפתח הפרטי הוא סודי (ורצוי שיהיה מאוחסן באופן לא מקוון, כלומר לא נגיש בשום דרך מהאינטרנט), בעוד המפתח הפומבי יכול להיות ידוע לכולם. עפ"י העקרונות של קריפוטגרפיה אסימטרית, המפתח הפרטי יכול לשמש את שרת השמות כדי לחתום על רשומות המשאב שהוא שולח, והמפתח הפומבי יכול לשמש כל גורם אחר לצורך וידוא החתימה. וידוא מוצלח של החתימה מהווה אישור לכך שהמפתח הפרטי המתאים הוא זה שביצע את החתימה (ובאופן זה מאמתים את זהות השרת), וכן אישור לכך שהתוכן שהתקבל על-ידי הגורם המוודא הוא אותו תוכן שעליו חתם השרת (ובאופן זה מובטחת שלמות התוכן, כלומר שאף אחד לא שינה אותו בדרך).

בגרסה המוקדמת של DNSSEC, שהוגדרה ב-1999 במסגרת RFC 2535, נקבע כי כל שרת שמות ב-DNS יהיה אחראי לחתום על שמות המתחם שמתחתיו בהיררכית ה-DNS. הגישה הזאת הפכה את הפרוטוקול ללא מעשי - למשל, המשמעות היא ששרת ה-DNS האחראי על com, למשל, יצטרך לחתום על כל רשומות ה-DNS של שמות המתחם שמתחתיו, מאמץ לא פשוט (עשרות מיליוני רשומות). עדכון של מפתחות החתימה הופך למשימה עצומה, מאחר ואז יש לחתום מחדש על רשומות ה-DNS של כל תתי-המתחם.

ניהול המפתחות של DNSSEC

הגרסה המעודכנת של DNSSEC (2005) הכניסה לשימוש היררכיה של מפתחות, שאפשרה לשרת DNSSEC להאציל סמכויות חתימה על שרתים ברמות נמוכות יותר. בגרסה זו, כל שרת DNS מנהל שני זוגות של מפתחות פרטיים/פומביים:

1. מפתחות המיועדים לחתימה על רשומות משאב שבאחריות השרת - Zone Signing Keys (ובקיצור ZSK).
2. מפתחות המיועדים לחתימה על מפתחות ZSK - Key Signing Keys (ובקיצור KSK).

לדוגמה, השרת האחראי על com מחזיק מפתח ZSK, שמשמש אותו לחתימה על רשומות משאב - למשל, רשומות המשאב המכילות את התרגום של Wikipedia.com לכתובת ה-IP המתאימה. ה-ZSK יכול לחתום גם על מפתחות KSK של שרתים ברמה נמוכה יותר בהיררכיה - למשל, אם לאתר ויקיפדיה יש מפתח KSK המשמש אותו ל-DNSSEC, אז שרת השמות של com יכול לחתום על מפתח זה, ובאופן זה לספק הוכחה לכך שאותו KSK אכן שייך לאתר ויקיפדיה. כפי שיתואר בהמשך, מפתחות הם תוכן שניתן להעביר אותו ברשומת משאב בדיוק באותו אופן ששרתי DNS מעבירים פרטים של כתובות IP או מידע אחר, ולכן גם ניתן לחתום על תוכן זה באופן דומה. בנוסף, השרת האחראי על com מחזיק גם מפתח KSK - מפתח זה ישמש אותו כדי לחתום על מפתח ה-ZSK של עצמו - עוד על תהליך זה בהמשך.

לגישה זו לניהול מפתחות יש שני יתרונות על-פני הגרסה הקודמת:

ראשית, כל שרת DNS יכול להחליף את מפתחות החתימה שלו (ZSK) ללא צורך לעדכן שום רשומות ברמות הגבוהות יותר בהיררכיה. למשל, אם ויקיפדיה יחליפו את ה-ZSK שלהם, זה ישפיע רק ברמה שלהם - החתימה של com על ה-KSK של ויקיפדיה עדיין בתוקף, והמנהלים של ויקיפדיה צריכים רק לייצר באופן מקומי חתימה על ה-ZSK החדש עם מפתח ה-KSK שלהם.

שנית, כל מי שרוצה לאמת רשומות DNSSEC, נדרש להחזיק רק את מפתח ה-KSK הפומבי של שרתי השורש: מפתח זה יכול לאמת רשומות בהן שרתי השורש מספקים את מפתח ה-ZSK הפומבי שלהם, כאשר הוא חתום על-ידי מפתח ה-KSK. מפתח ה-ZSK הפומבי, בתורו, יכול לשמש כדי לאמת חתימה של שרתי השורש על מפתח KSK פומבי של שרת DNSSEC ברמה נמוכה יותר בהיררכיה (למשל ה-KSK הפומבי של com), וכן הלאה.

סוגי רשומות חדשים ב-DNSSEC

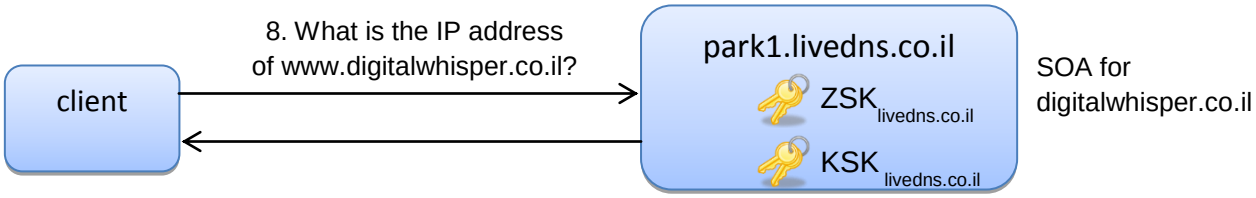
כפי שכבר צויין, פרוטוקול DNS יודע לנהל סוגים שונים של רשומות משאב, כמו רשומות מסוג "A" (עבור תרגום שמות מתחם לכתובות IP) ומסוג "NS" (עבור קבלת שם שרת ה-DNS האחראי על שם מתחם). כדי לתמוך ב-DNSSEC, הוגדרו רשומות משאב נוספות:

סוג	תיאור
RRSIG	Resource Record Signature: רשומה זו מכילה חתימה על אוסף של רשומות משאב אחרות הנשלחות בתשובת DNS.
DNSKEY	DNS public key: רשומה זו מתארת מפתח פומבי המשמש ב-DNSSEC.
DS	Delegation Signer: רשומה זו משמשת לאימות רשומת DNSKEY של שם המתחם הבא בהיררכית DNS. היא מכילה תמצית (hash) קריפטוגרפית של המפתח הפומבי של שם המתחם אותו מאמתים. לכן שליחת רשומת DS ביחד עם רשומת RRSIG מתאימה המכילה חתימה חוקית שלה, מהווה הוכחה למהימנות המפתח הפומבי.
NSEC	Next Secure: משמשת לציין טווח של שמות מתחם שאינם קיימים. DNSSEC מאמת גם תשובות שליליות ("הכתובת ששאלת עליה לא קיימת"). הטיפול בתשובות שליליות אינו טריוויאלי, ויש לו גם השלכות מבחינת פרטיות, אולם פירוט הטיפול ב-NSEC חורג מהיקף כתבה זו.

כדי להבהיר את תפקידן של רשומות המשאב החדשות וכיצד הן משתלבות ב-DNS, נראה תשובת DNSSEC לדוגמה.

דוגמה לתשובה חתומה עם DNSSEC

כדי להמחיש כיצד DNSSEC עובד, וכיצד סוגי הרשומות החדשים באים לידי ביטוי, נניח כי במהלך תהליך התרגום של הכתובת www.digitalwhisper.co.il נשלחת שאילתת DNSSEC לשרת park1.livedns.co.il



(מהדוגמה הקודמת).

תשובת DNSSEC שתישלח מ-park1.livedns.co.il בחזרה לשולח השאילתא, עשויה להכיל את הרשומות הבאות (ההסבר מתחת לטבלה. הכל ספקולטיבי כמובן, שמות המתחם שמדובר עליהם עדיין לא תומכים ב-DNSSEC):

		Name	Data	TTL
[1]	An:	www.digitalwhisper.co.il	A = 193.105.99.162	4 hrs
	An:	www.digitalwhisper.co.il	$RRSIG_{(A)} = SIG_{ZSKlivedns.co.il}$([1])	4 hrs
[2]	Au:	digitalwhisper.co.il	NS = park1.livedns.co.il	4 hrs
	Au:	digitalwhisper.co.il	NS = park2.livedns.co.il	4 hrs
	Au:	digitalwhisper.co.il	$RRSIG_{(NS)} = SIG_{ZSKlivedns.co.il}$([2])	4 hrs
[3]	Ad:	park1.livedns.co.il	A = 62.219.78.217	4 hrs
[4]	Ad:	park2.livedns.co.il	A = 118.139.160.111	4 hrs
	Ad:	park1.livedns.co.il	$RRSIG_{(A)} = SIG_{ZSKlivedns.co.il}$([3])	4 hrs
	Ad:	park2.livedns.co.il	$RRSIG_{(A)} = SIG_{ZSKlivedns.co.il}$([4])	4 hrs

שדות ה-TTL מתארים את "אורך החיים" של כל אחת מהתשובות, ובפרט קובעים כמה זמן ניתן לשמור את התשובה בזכרון המטמון של השרת. ה-TTL נקבע על-ידי המנהלים של כל שם מתחם.

[1] הרשומה הראשונה היא רשומה מסוג A (תרגום שם מתחם לכתובת IP), המכילה את התשובה (**An**swer) לשאילתה, כלומר כתובת ה-IP עבור שם המתחם שעליו נשאלה השאלה (www.digitalwhisper.co.il). אל הרשומה מצורפת רשומה נוספת מסוג RRSIG, המכילה חתימה על רשומת ה-A עם מפתח ה-ZSK של livedns.co.il.

[2] שתי הרשומות הבאות הן רשומות מסוג NS (Name Server), המספקות מידע לגבי שרתי ה-DNS האחראיים (**Au**thoritative) על digitalwhisper.co.il. במקרה זה, מסופקות שתי חלופות (שני שרתים). אליהם מצורפת רשומה נוספת מסוג RRSIG, המכילה חתימה על שתי רשומות ה-NS עם מפתח ה-ZSK של livedns.co.il.

[3] הרשומה הבאה היא רשומה של מידע נוסף (**Ad**ditional), כלומר "רשומת דבק". היא מספקת את כתובת ה-IP עבור park1.livedns.co.il, ששמו נמסר ב-[2].

[4] רשומת דבק נוספת מספקת גם את כתובת ה-IP עבור park2.livedns.co.il. שתי רשומות הדבק מלוות על-ידי שתי רשומות חתימה, כאשר החתימות הן עם מפתח ה-ZSK של livedns.co.il.

ככלל, מסופקת רשומת RRSIG לכל אוסף רשומות משאב עם אותו שם (Name, כמו digitalwhisper.co.il לעיל), סוג (Type, כמו "A" או "NS" בדוגמה), ומחלקה (Class, לרוב יהיה IN עבור Internet). למשל, שתי רשומות ה-NS נחתמות ביחד מאחר והן מאותו סוג ומתייחסות לאותו שם. רשומות המידע הנוסף, לעומת זאת, מתייחסות לשני שמות שונים (park1 ו-park2), ולכן יש רשומת חתימה נפרדת לכל אחת מהן.

כאשר הלקוח מקבל את התשובה הנ"ל, במידה ויש ברשותו את מפתח ה-ZSK הפומבי של livedns.co.il, ביכולתו לוודא את כל החתימות שברשומות ה-RRSIG, ולדעת כי המידע ברשומות המשאב אכן נשלח על-ידי השרת livedns.co.il (אימות) ולא עבר שינוי בדרך (שלמות).

אבל מה אם אין ללקוח את מפתח ה-ZSK הפומבי של livedns.co.il?

קבלת מידע על מפתחות עם שאילתות DNSKEY

במקרה כזה, ניתן לשלוח שאילתת DNSSEC המבקשת רשומה מסוג DNSKEY עבור park1.livedns.co.il. בתהליך זה, הלקוח יקבל תשובה שעשויה להיראות כך:

		Name	Data	TTL
[1]	An:	park1.livedns.co.il	$DNSKEY_{(ZSK)}$ = $PUB\_ZSK_{livedns.co.il}$	4 hrs
	An:	park1.livedns.co.il	$DNSKEY_{(KSK)}$ = $PUB\_KSK_{livedns.co.il}$	4 hrs
	An:	park1.livedns.co.il	$RRSIG_{(DNSKEY)}$ = $SIG_{KSKlivedns.co.il}$([1])	4 hrs

התשובה מכילה את שני המפתחות הפומביים של livedns.co.il: מפתח ה-ZSK (שבאמצעותו ניתן לוודא את החתימות על רשומות המשאב מהתשובה הקודמת), ומפתח ה-KSK. התשובה חתומה עם מפתח ה-KSK הפרטי של livedns.co.il, וכיוון שמפתח ה-KSK הפומבי נתון בתשובה, ניתן מיד לאמת את החתימה.

יש לשים לב שמטרתה העיקרית של שאילתת ה-DNSKEY הייתה לספק את מפתח ה-ZSK של השרת. לצורך האימות נעשה שימוש במפתח ה-KSK, שגם סופק בתשובה. זה מצב בעייתי - אם אין כבר בידינו את ה-KSK, אז יש כאן מעגליות - אנחנו צריכים להאמין למידע שסופק בתשובה בשביל שנוכל לבדוק את אמינות המידע שבתשובה. בפרט, כל גורם זדוני היה יכול לייצר מפתח KSK משלו שלכאורה שייך ל-park1.livedns.co.il, ולהשתמש בו כדי לחתום על שאילתת ה-DNSKEY. כדי לבסס את האמון במפתח KSK, צריך לשאול גורם אחר שסומכים עליו (בדרך-כלל גורם שנמצא מעל park1.livedns.co.il בהיררכית ה-DNS), ויכול לערוב שמפתח ה-KSK שברשותנו נכון.

ביסוס אמון עם שאילתות DS

כדי לבסס את האמון במפתחות KSK, ניתן להשתמש בשאילתות על רשומות DS (Delegation Signer). רשומות אלה מכילות את התמצית הקריפטוגרפית של מפתח ה-KSK, ויחתום עליהן הגורם שתת-התחום park1.livedns.co.il נמצא תחתיו, במקרה זה co.il. בתהליך ביצוע שאילתת DNSSEC לקבלת מידע DS עבור park1.livedns.co.il, עשויה להתקבל התשובה הבאה משרת שמות ה-DNS של co.il:

		Name	Data	TTL
[1]	An:	park1.livedns.co.il	DS = $PUB\_KSK_{livedns.co.il}$	7 days
	An:	park1.livedns.co.il	$RRSIG_{(DS)}$ = $SIG_{ZSKco.il}$([1])	7 days

שרת השמות של co.il שולח נתונים על מפתח ה-KSK הפומבי של livedns.co.il (שניתן להצליב עם המידע שהתקבל בתשובת ה-DNSKEY שהתקבלה לפני-כן מ-livedns.co.il), וחותם על הרשומה עם מפתח ה-ZSK הפרטי שלו. כדי לאמת את החתימה, נשתמש במפתח ה-ZSK הפומבי של co.il, במידה והוא ברשותנו. במידה ולא, חוזרים על תהליך דומה - שולחים שאילתת DNSKEY עבור co.il כדי לקבל את מפתחות ה-ZSK וה-KSK הפומביים של המתחם, ולאחר-מכן שולחים שאילתת DS כדי לבסס אמון במפתח ה-KSK. הפעם נקבל תשובה משרתי השורש של DNS, הנמצאים מעל co.il בהיררכית ה-DNS. מפתחות ה-KSK של שרתי השורש מהווים **שורש האמון** - כל השירותים התומכים באימות DNSSEC צריכים להכיר אותם (למשל על-ידי קונפיגורציה המתבצעת על-ידי מנהל המערכת) כדי שתהיה נקודת פתיחה ממנה ניתן לבסס את האמון בשאר המפתחות במערכת.

נחזור בנקודה זו על השאלה לגבי הצורך בפיצול בין מפתחות ה-ZSK ומפתחות ה-KSK - הרי באותה מידה, שרת השמות של co.il היה יכול לחתום ישירות על מפתח ה-ZSK של livedns.co.il, במקום להוסיף עוד חוליה בשרשרת העוברת דרך מפתח ה-KSK של livedns.co.il. הסיבה שעושים הפרדה בין המפתחות היא כדי להחליש את התלויות בין הרמות השונות בהיררכיה. למשל, המנהלים של livedns.co.il יכולים להחליף את מפתחות ה-ZSK שלהם, ולייצר מחדש חתימות לכל תתי-המתחם שברשותם, אולם זה לא ידרוש שום מעורבות מצד המנהלים של co.il - מפתח ה-KSK של livedns.co.il והחתימה עליו עדיין יהיו תקפים, ומנהלי livedns.co.il צריכים רק לייצר חתימה חדשה על מפתח ה-ZSK החדש עם ה-KSK שברשותם כדי להפוך אותו למפתח לגיטימי. ההפרדה בין המפתחות הופכת את המערכת להרבה יותר סקלבילית, והיא אחד ההבדלים המשמעותיים בין סדרת תקני ה-DNSSEC מ-2005 ובין התקנים הקודמים.

סיכום

פרוטוקול DNS מהווה את אחת מאבני הבניין הבסיסיות ביותר של רשת האינטרנט. עם זאת, הוא תוכנן בימיה הראשונים של רשת האינטרנט, בזמן שבעיות אבטחת מידע לא היו ממש על הפרק. פרוטוקול DNSSEC מיועד להרחיב את DNS בצורה שתשפר את בטיחות הפרוטוקול תוך התבססות על מפתחות קריפטוגרפיים. אולם תהליך ההטמעה של DNSSEC אינו פשוט ומתקדם באיטיות, דווקא בגלל החלק המרכזי של DNS בפעילות התקינה של האינטרנט, והחשש ששינויים יערערו את המערכת. ישנה גם בעיית "ביצה ותרנגולת" - קשה להצדיק את ההשקעה הכרוכה בהטמעת DNSSEC ברמת השרת כל עוד אין לקוחות המסוגלים לבצע אימות DNSSEC, ואין טעם לבצע אימות DNSSEC כל עוד אין שרתים השולחים תשובות חתומות. קושי נוסף, נובע מכך שההטמעה מתבצעת בקצב שונה במדינות שונות. למשל,

מפתחות עבור שרתי השורש נעשו זמינים רק ב-2010, בעוד מימושים של הפרוטוקול החלו לפעול עוד ב-2005, כך שנדרשו פתרונות אחרים כדי לתת מענה לבעיית שורש האמון.

עם כל קשיי ההטמעה, בעיות אבטחה הנוגעות לפרוטוקול DNS, כמו ההתקפה שדן קמינסקי הציג ב-2008, מהוות תזכורת לגבי החשיבות של פתרון כמו DNSSEC, ונותנות דחיפה לתהליך ההטמעה. כאשר DNSSEC ייפרס בצורה רחבה יותר, יוכל לשמש גם כתשתית לניהול מידע קריפטוגרפי באינטרנט, עם שימושים אפשריים כגון העברת מפתחות לצורך SSH או IPsec, או הטמעה של אימות עבור מערכות דואר אלקטרוניות, תוך שימוש בתשתית DNSSEC להעברת המפתחות.

מקורות ומידע נוסף

1. קל להריץ ולראות שאילתות DNSSEC "חיות" באמצעות אתרים המספקים שירותי חיפוש DNS. לדוגמה, באתר http://centralops.net/co/NSLookup.aspx ניתן לבחור לבצע שאילתות מסוג DNSKEY או DS עבור שמות מתחם התומכים ב-DNSSEC, כמו com או org.
2. מידע כללי על DNS:

DNS and BIND, O'Reilly http://shop.oreilly.com/product/9780596100575.do

3. מבוא ל-DNSSEC:

A Fundamental look at DNSSEC, Deployment and DNS Security Extensions, by Geoff Huston
http://www.circleid.com/posts/dnssec_deployment_and_dns_security_extensions/URL%202

4. מידע סטטיסטי על פריסת DNSSEC:

Counting DNSSEC, by Geoff Huston and George Michaelson
http://www.potaroo.net/ispcol/2012-10/counting-dnssec.html

Recounting DNSSEC, by Geoff Huston and George Michaelson
http://www.potaroo.net/ispcol/2012-10/counting-dnssec-2.html

5. שקפים בנושא DNSSEC מכנס NANOG 51:

http://www.nanog.org/meetings/nanog51/presentations/Sunday/DNSSEC-tutorial-for-NANOG51-2011-01.pdf

6. מאמרים נוספים ב-DigitalWhisper הנוגעים ל-DNS:
   - גליון 2, נובמבר 2009: DNS Cache Poisoning, מאת אפיק קסטיאל.

http://www.digitalwhisper.co.il/files/Zines/0x02/DW2-7-DNS-Cache-Poisoning.pdf

- גליון 9, יוני 2010: DNS Rebinding, מאת אביעד (greenblast).
  http://www.digitalwhisper.co.il/files/Zines/0x09/DW9-3-DNSRebind.pdf
- גליון 18, מרץ 2011: Domain Name System - אנומליות, איתור ומניעה, מאת קיריל לשצ'יבר.
  http://digitalwhisper.co.il/files/Zines/0x12/DW18-4-DNS.pdf
- גליון 25, אוקטובר 2011, DNS Cache Snooping, מאת עוז אליסיאן.
  http://www.digitalwhisper.co.il/files/Zines/0x19/DW25-5-DNSSnooping.pdf

על המחבר

ד"ר אריק פרידמן עובד כחוקר במכון המחקר NICTA בסידני, אוסטרליה. תחומי המחקר שלו מתמקדים בפרטיות ואבטחת מידע, ובעיקר בשילובם במסגרת אלגוריתמים ללמידה ממוחשבת וכריית נתונים. אריק סיים את לימודי הדוקטורט בפקולטה למדעי המחשב בטכניון בשנת 2011, והוא מחזיק גם בתואר MBA מאוניברסיטת תל-אביב.